Micro Fiche Scan

Name of device(s) tested:

DEUNA

Test description:

REPAIR DIAG

MAINDEC Number or Package Identifier (after SEP 1977):

CZUAAB0

Fiche Document Part Number:

AH-T364B-MC

Fiche preparation date unknown, using copyright year:

1983

Image resolution:

8-bit gray levels, max. quality for archiving

.REM &

IDENTIFICATION

PRODUCT CODE: AC-T363B-MC

PRODUCT NAME: CZUAABO REPAIR DIAG

PRODUCT DATE: JANUARY 1983

MAINTAINER: DISTRIBUTED SYSTEMS DIAGNOSTIC ENGINEERING

AUTHOR: MICHAEL CINNAMON

TABLE OF CONTENTS

```
.PAGE
&
.REM &
```

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

THIS PRODUCT IS THE PDP-11 REPAIR LEVEL DIAGNOSTIC FOR THE UNIBUS TO NI ADAPTER (DEUNA). THIS DIAGNOSTIC WAS DESIGNED TO DETECT STATIC AND DYNAMIC HARDWARE FAILURES IN THE DEUNA BOARDSET. THE DEUNA BOARDSET IS THE TWO MODULES WHICH PLUG INTO THE PDP-11 UNIBUS. THE TWO MODULES ARE THE M7792 PORT MODULE AND THE M7793 LINK MODULE. THIS DIAGNOSTIC IS CAPABLE OF TESTING EIGHT SUCH BOARDSETS ON A SINGLE PDP-11 UNIBUS.

THIS DIAGNOSITC WILL ONLY RUN IN A STANDALONE, OFFLINE ENVIRONMENT. THE DEUNA IS LOGICALLY REMOVED FROM THE 'WIRE' BY THE DIAGNOSTIC, SO NO MESSAGES FROM OTHER NODES ON THE NETWORK, TO THE DEUNA UNDER TEST, WILL DISRUPT THE TESTING PROCESS. HOWEVER, BECAUSE THIS DIAGNOSTIC RUNS THE DEUNA SELF-TEST IN TEST 9, AND THE SELF-TEST PERFORMS AN EXTERNAL LOOPBACK AS PART OF ITS TESTING PROCEDURE, IT IS RECOMMENDED THAT THE DEUNA TRANSCEIVER CABLE BE REMOVED FROM THE H4000 TRANSCEIVER AND PLUGGED INTO A FIELD SERVICE EXTERNAL LOOPBACK CONNECTOR.

THIS DIAGNOSTIC HAS BEEN WRITTEN FOR USE WITH THE DIAGNOSTIC RUNTIME SERVICES SOFTWARE (SUPERVISOR). THESE SERVICES PRUVIDE THE INTERFACE TO THE OPERATOR AND TO THE SOFTWARE ENVIRONMENT. THIS PROGRAM CAN BE USED WITH XXDP+, ACT, APT, SLIDE AND PAPER TAPE. FOR A COMPLETE DESCRIPTION OF THE RUNTIME SERVICES, REFER TO THE XXDP+ USER'S MANUAL. THERE IS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES IN SECTION 2 OF THIS DOCUMENT.

1.2 SYSTEM REQUIREMENTS

THE FOLLOWING LIST OF HARDWARE IS REQUIRED TO RUN THIS DIAGNOSTIC:

```
PDP-11 CPU
28K WORDS OF MEMORY
CONSOLE TERMINAL
DEUNA BOARDSET (M7792, M7793)
PLUS, ONE OF THE FOLLOWING:
        -LINK BOARD TO BULKHEAD CABLE CONNECTED AND BULKHEAD TO
          TRANSCEIVER TAP CABLE CONNECTED (NORMAL ONLINE CONFIGURATION)
        OR
        -LINK BOARD TO BULKHEAD CABLE CONNECTED AND BULKHEAD TO FIELD
          SERVICE EXTERNAL LOOPBACK CONNECTOR INSTALLED (OFFLINE CONFIGURATION)
```

1.3 RELATED DOCUMENTS AND STANDARDS

XXDP+ USERS MANUAL CHQUS
XXDP+ PROGRAMMERS MANUAL
DEUNA LINK BOARD FUNCTIONAL SPECIFICATION
DEUNA PORT BOARD FUNCTIONAL SPECIFICATION
DEUNA PROGRAMMING SPECIFICATION

1.4 DIAGNOSTIC HIERARCY PREREQUISITES

THIS DIAGNOSTIC ASSUMES THAT THE PDP-11 PROCESSOR AND MEMORY ARE IN WORKING CONDITION AND IS CAPABLE OF EXECUTING PDP-11 INSTRUCTIONS NORMALLY. THE UNIBUS IS EXPECTED TO BE FULLY OPERATIONAL I.E. ANY PROBLEMS REPORTED BY THIS DIAGNOSTIC, ABOUT THE INTEGRITY OF THE UNIBUS, ARE ASSUMED TO BE THE RESULT OF A FAILURE ON THE DEUNA AND NOT THE FAULT OF OTHER DEVICES CONNECTED TO THE UNIBUS.

THIS DIAGNOSTIC DOES NOT REQUIRE ANY PRELIMINARY TESTS BE EXECUTED ON THE DEUNA, NOR DOES RUNNING OF ANY OTHER TESTS PRIOR TO RUNNING THIS DIAGNOSTIC, AFFECT THE OPERATION OF THE TESTS IN THIS DIAGNOSTIC.

FOR A COMPLETE TEST OF THE DEUNA, ALL THE AVAILABLE DIAGNOSTIC SOFTWARE SHOULD BE RUN. THIS WOULD INCLUDE RUNNING THE DEUNA FUNCTIONAL DIAGNOSTIC AND THE DECX-11 SYSTEM EXERCISOR WITH THE DEUNA MODULE SELECTED.

1.5 ASSUMPTIONS

THIS DIAGNOSTIC ASSUMES THAT THE DEUNA WILL NOT HANG THE UNIBUS WHEN AN ACCESS IS MADE TO ANY ONE OF THE PCSR REGISTERS. THE DEUNA IS CAPABLE OF ASSERTING ACLO ON THE UNIBUS, THIS FEATURE COULD, IF BROKEN, CAUSE THE UNIBUS TO HANG. THIS TYPE OF FAILURE IS NOT DETECTED BY THE DIAGNOSTIC.

PORTIONS OF THIS DIAGNOSTIC USE SPECIAL DIAGNOSTIC MICROCODE THAT IS LOADED INTO THE DEUNA WRITEABLE CONTROL STORE. THIS MICROCODE ALLOWS THE DIAGNOSTIC MORE VISABILITY INTO THE INTERNALS OF THE DEUNA HARDWARE AS WELL AS NOT RELYING AS HEAVILY ON THE COMPLEX OPERATIONAL MICROCODE IN ROM, HOWEVER, THIS INCREASES THE DIAGNOSTIC'S COMPLEXITY SOMEWHAT. THEREFORE, IT IS ASSUMED THAT THE USER OF THIS DIAGNOSTIC IS FAMILIAR WITH THE DEUNA ENOUGH TO READ DEUNA MICROCODE SHOULD AN ERROR OCCUR.

2.0 OPERATING INSTRUCTIONS

THIS SECTION CONTAINS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES. FOR DETAILED INFORMATION, REFER TO THE XXDP+ USER'S MANUAL (CHQUS).

2.1 COMMANDS

THERE ARE ELEVEN LEGAL COMMANDS FOR THE DIAGNOSTIC RUNTIME SERVICES (SUPERVISOR). THIS SECTION LISTS THE COMMANDS AND GIVES A VERY BRIEF DESCRIPTION OF THEM. THE XXDP+ USER'S MANUAL HAS MORE DETAILS.

COMMAND	EFFECT
START	START THE DIAGNOSTIC FROM AN INITIAL STATE
RESTART	START THE DIAGNOSTIC WITHOUT INITIALIZING
CONTINUE	CONTINUE AT BTEST THAT WAS INTERRUPTED (AFTER ^C)
PROCEED	CONTINUE FROM AN ERROR HALT
EXIT	RETURN TO XXDP+ MONITOR (XXDP+ OPERATION ONLY!)
ADD	ACTIVATE A UNIT FOR TESTING (ALL UNITS ARE CONSIDERED TO BE ACTIVE AT START TIME
DROP	DEACTIVATE A UNIT

```
        DISPLAY         TYPE A LIST OF ALL DEVICE INFORMATION
        FLAGS           TYPE THE STATE OF ALL FLAGS (SEE SECTION 2.3)
        ZFLAGS          CLEAR ALL FLAGS (SEE SECTION 2.3)
```

A COMMAND CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. SO YOU MAY, FOR EXAMPLE. TYPE "STA" INSTEAD OF "START".

2.2 SWITCHES

THERE ARE SEVERAL SWITCHES WHICH ARE USED TO MODIFY SUPERVISOR OPERATION. THESE SWITCHES ARE APPENDED TO THE LEGAL COMMANDS. ALL OF THE LEGAL SWITCHES ARE TABULATED BELOW WITH A BRIEF DESCRIPTION OF EACH. IN THE DESCRIPTIONS BELOW, A DECIMAL NUMBER IS DESIGNATED BY "DDDDD".

```
        SWITCH          EFFECT
        ------          ---------------------------------
        /TESTS:LIST     EXECUTE ONLY THOSE TESTS SPECIFIED IN
                        THE LIST.  LIST IS A STRING OF TEST
                        NUMBERS, FOR EXAMPLE - /TESTS:1:5:7-10.
                        THIS LIST WILL CAUSE TESTS 1,5,7,8,9,10 TO
                        BE RUN.  ALL OTHER TESTS WILL NOT BE RUN.
        /PASS:DDDDD     EXECUTE DDDDD PASSES (DDDDD = 1 TO 64000)
        /FLAGS:FLGS     SET SPECIFIED FLAGS.  FLAGS ARE DESCRIBED
                        IN SECTION 2.3.
        /EOP:DDDDD      REPORT END OF PASS MESSAGE AFTER EVERY
                        DDDDD PASSES ONLY.  (DDDDD = 1 TO 64000)
        /UNITS:LIST     TEST/ADD/DROP ONLY THOSE UNITS SPECIFIED
                        IN THE LIST.  LIST EXAMPLE - /UNITS:0:5:10-12
                        USE UNITS 0,5,10,11,12 (UNIT NUMBERS = 0-63)
```

EXAMPLE OF SWITCH USAGE:

```
        START/TESTS:1-5/PASS:1000/EOP:100
```

THE EFFECT OF THIS COMMAND WILL BE: 1) TESTS 1 THROUGH 5 WILL BE EXECUTED, 2) ALL UNITS WILL TESTED 1000 TIMES AND 3) THE END OF PASS MESSAGES WILL BE PRINTED AFTER EACH 100 PASSES ONLY. A SWITCH CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. YOU MAY, FOR EXAMPLE, TYPE "/TES:1-5" INSTEAD OF "/TESTS:1-5".

BELOW IS A TABLE THAT SPECIFIES WHICH SWITCHES CAN BE USED BY EACH COMMAND.

	TESTS	PASS	FLAGS	EOP	UNITS
START	X	X	X	X	X
RESTART	X	X	X	X	X
CONTINUE		X	X	X	
PROCEED			X		
DROP					X
ADD					X
DISPLAY					X
FLAGS					
ZFLAGS					
EXIT					

2.3 FLAGS

FLAGS ARE USED TO SET UP CERTAIN OPERATIONAL PARAMETERS SUCH AS LOOPING ON ERROR. ALL FLAGS ARE CLEARED AT STARTUP AND REMAIN CLEARED UNTIL EXPLICITLY SET USING THE FLAGS SWITCH. FLAGS ARE ALSO CLEARED AFTER A START COMMAND UNLESS SET USING THE FLAG SWITCH. THE ZFLAGS COMMAND MAY ALSO BE USED TO CLEAR ALL FLAGS. WITH THE EXCEPTION OF THE START AND ZFLAGS COMMANDS, NO COMMANDS AFFECT THE STATE OF THE FLAGS; THEY REMAIN SET OR CLEARED AS SPECIFIED BY THE LAST FLAG SWITCH.

FLAG	EFFECT
HOE	HALT ON ERROR - CONTROL IS RETURNED TO RUNTIME SERVICES COMMAND MODE
LOE	LOOP ON ERROR
IER*	INHIBIT ALL ERROR REPORTS
IBE*	INHIBIT ALL ERROR REPORTS EXCEPT FIRST LEVEL (FIRST LEVEL CONTAINS ERROR TYPE, NUMBER, PC, TEST AND UNIT)
IXE*	INHIBIT EXTENDED ERROR REPORTS (THOSE CALLED BY PRINTX MACRO'S)
PRI	DIRECT MESSAGES TO LINE PRINTER
PNT	PRINT TEST NUMBER AS TEST EXECUTES
BOE	"BELL" ON ERROR
UAM	UNATTENDED MODE (NO MANUAL INTERVENTION)
IDR	INHIBIT PROGRAM DROPPING OF UNITS
ADR	EXECUTE AUTODROP CODE
LOT	LOOP ON TEST

*ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1

SEE THE XXDP+ USER'S MANUAL FOR MORE DETAILS ON FLAGS. YOU MAY SPECIFY MORE THAN ONE FLAG WITH THE FLAG SWITCH. FOR EXAMPLE, TO CAUSE THE PROGRAM TO LOOP ON ERROR, INHIBIT ERROR REPORTS AND TYPE A "BELL" ON ERROR, YOU MAY USE THE FOLLOWING STRING:

```
/FLAGS:LOE:IER:BOE
```

2.4 HARDWARE QUESTIONS

WHEN A DIAGNOSTIC IS STARTED, THE RUNTIME SERVICES WILL PROMPT THE USER FOR HARDWARE INFORMATION BY TYPING "CHANGE HW (L) ?" YOU MUST ANSWER "Y" AFTER A START COMMAND UNLESS THE HARDWARE INFORMATION HAS BEEN "PRELOADED" USING THE SETUP UTILITY (SEE CHAPTER 6 OF THE XXDP+ USER'S MANUAL). WHEN YOU ANSWER THIS QUESTION WITH A "Y", THE RUNTIME SERVICES WILL ASK FOR THE NUMBER OF UNITS (IN DECIMAL). YOU WILL THEN BE ASKED THE FOLLOWING QUESTIONS FOR EACH UNIT.

WHAT IS THE PCSRO ADDRESS?

```
                THIS IS THE ADDRESS AT WHICH PCSRO RESIDES ON THE UNIBUS.
                THIS ADDRESS IS SWITCH SELECTABLE ON THE PORT MODULE.
                THE ALLOWABLE RANGE IS 160000-177776.

        WHAT IS THE VECTOR ADDRESS?
                THIS IS THE INTERRUPT VECTOR ADDRESS. THIS ADDRESS IS ALSO
                SWITCH SELECTABLE ON THE PORT MODULE. THE ALLOWABLE RANGE
                IS 000-776.

        SAMPLE DIALOGUE:

        DR>START

        CHANGE HW (L)  ? Y

        # UNITS (D)  ? 2

        UNIT 0
        WHAT IS THE PCSRO ADDRESS? (O)  ? 170000
        WHAT IS THE VECTOR ADDRESS? (O)  ? 700

        UNIT 1
        WHAT IS THE PCSRO ADDRESS? (O)  ? 170010
        WHAT IS THE VECTOR ADDRESS? (O)  ? 710


        2.7 QUICK START-UP PROCEDURE  (XXDP+)

        TO START-UP THIS PROGRAM:

                1. BOOT XXDP+

                2. GIVE THE DATE AND ANSWER ANY QUESTIONS THE MONITOR ASKS

                3. TYPE "R NAME", WHERE NAME IS THE NAME OF THE BIN OR BIC
                   FILE FOR THIS PROGRAM

                4. TYPE "START"

                5. ANSWER THE "CHANGE HW" QUESTION WITH "Y"

                6. ANSWER ALL THE HARDWARE QUESTIONS

        WHEN YOU FOLLOW THIS PROCEDURE YOU WILL BE USING ONLY THE
        DEFAULTS FOR FLAGS AND SOFTWARE PARAMETERS.  THESE DEFAULTS
        ARE DESCRIBED IN SECTIONS 2.3 AND 2.5.

        3.0 ERROR INFORMATION

        3.1 TYPES OF ERROR MESSAGES

        THERE ARE THREE LEVELS OF ERROR MESSAGES THAT MAY BE ISSUED BY
        A DIAGNOSTIC: GENERAL, BASIC AND EXTENDED.  GENERAL ERROR MESSAGES
        ARE ALWAYS PRINTED UNLESS THE "IER" FLAG IS SET (SECTION 2.3).
        THE GENERAL ERROR MESSAGE IS OF THE FORM:
```

```
          NAME  TYPE  NUMBER  ON UNIT NUMBER  TST NUMBER PC:XXXXXX
          ERROR MESSAGE

.WHERE;   NAME = DIAGNOSTIC NAME
          TYPE = ERROR TYPE (SYS FATAL, DEV FATAL, HARD OR SOFT)
          NUMBER = ERROR NUMBER
          UNIT NUMBER = 0 - N (N IS LAST UNIT IN PTABLE)
          TST NUMBER = TEST AND SUBTEST WHERE ERROR OCCURRED
          PC:XXXXXX = ADDRESS OF ERROR MESSAGE CALL
```

BASIC ERROR MESSAGES ARE MESSAGES THAT CONTAIN SOME ADDITIONAL INFORMATION ABOUT THE ERROR. THESE ARE ALWAYS PRINTED UNLESS THE ''IER'' OR ''IBR'' FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL MESSAGE.

EXTENDED ERROR MESSAGES CONTAIN SUPPLEMENTARY ERROR INFORMATION SUCH AS REGISTER CONTENTS OR GOOD/BAD DATA. THESE ARE ALWAYS PRINTED UNLESS THE ''IER'', ''IBE'' OR ''IXE'' FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL ERROR MESSAGE AND ANY ASSOCIATED BASIC ERROR MESSAGES.

4.0 PERFORMANCE AND PROGRESS REPORTS

AT THE END OF EACH PASS, THE PASS COUNT IS GIVEN ALONG WITH THE TOTAL NUMBER OF ERRORS REPORTED SINCE THE DIAGNOSTIC WAS STARTED. THE ''EOP'' SWITCH CAN BE USED TO CONTROL HOW OFTEN THE END OF PASS MESSAGE IS PRINTED. SECTION 2.2 DESCRIBES SWITCHES.

5.0 DEVICE INFORMATION TABLES

AT THE COMPLETION OF THE FIRST PASS OF EACH DEUNA BEING TESTED, INFORMATION FOR THAT DEUNA IS PRINTED. THIS PRINTOUT CONTAINS THE ETHERNET DEFAULT ADDRESS (OBTAINED BY READING THE PHYSICAL ADDRESS ROM) , THE OPERATIONAL MICROCODE ROM VERSION NUMBER, AND THE SWITCH PACK SETTINGS FOR SELF TEST LOOPING AND REMOTE BOOTING.

EXAMPLE PRINTOUT:

```
ETHERNET DEFAULT ADDRESS (HEX): AA-00-03-00-00-02

ROM MICROCODE VERSION (DECIMAL:  1

SWITCH PACK SET FOR :

        SELF TEST LOOP DISABLED

        REMOTE BOOT ENABLED
```

6.0 TEST SUMMARIES

TEST 1: PCSR0 READ ACCESS TEST

THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER 0 CAN BE READ FROM THE UNIBUS AND THAT THE PREDETERMINED BITS APPEAR IN THE EXPECTED BIT POSITIONS.

TEST 2: PCSR1 READ ACCESS TEST

THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER CAN BE READ FROM THE UNIBUS AND THAT THE PREDETERMINED BITS APPEAR IN THE EXPECTED BIT POSITIONS.

TEST 3: PCSR2 READ ACCESS TEST

THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER 2 CAN BE READ FROM THE UNIBUS

TEST 4: PCSR3 READ ACCESS TEST

THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER 3 CAN BE READ FROM THE UNIBUS

TEST 5: RESET TEST

THIS TEST WILL VERIFY THE RESET STATE FOR ALL DEUNA UNIBUS REGISTERS

TEST 6: PCSR2 REGISTER READ/WRITE TEST

THIS TEST WILL CHECK THE REGISTER FOR ALL SA0 AND SA1 ERRORS (STUCK AT 0 AND STUCK AT 1 ERRORS). THE HOST WILL WRITE PATTERNS TO THE REGISTER AND READ THEM BACK TO VERIFY. THE PATTERNS TO BE USED ARE AT THE LABEL PATERN:: IN THE GLOBAL DATA SECTION OF THIS PROGRAM.

NOTE: SINCE PCSR2 BIT 00 IS ALWAYS PRESET TO LOGIC 0, THE LOWEST ORDER BIT OF THE PATTERN WILL BE MASKED BEFORE DOING THE COMPARISON.

TEST 7: REGISTER PCSR3 READ/WRITE TEST

THIS TEST WILL WRITE PATTERNS TO THE WRITEABLE FIELD OF PCSR3 AND WILL READ THESE BACK FOR VERIFICATION.

TEST 8: NOP TEST

THIS TEST WILL VERIFY THAT THE DEUNA PROCESSOR IS ALIVE AND CAN RESPOND TO A PORT COMMAND ISSUED. THE NOP PORT COMMAND WILL BE ISSUED TO THE DEUNA IN PCSR0 BITS 3:0 AND WILL WAIT FOR THE 'DNI' BIT TO SET IN PCSR0.

THE NOP PORT COMMAND USES A MINIMUM OF HARDWARE BUT FORCES THE T11 TO EXECUTE THE PORT COMMAND SEQUENCE.

TEST 9: SELF TEST

THIS TEST VERIFIES THAT THE ROM BASED SELF TEST
CAN BE RUN SUCCESSFULLY WHEN INVOKED VIA
THE SELF TEST PORT COMMAND.

TEST 10: DEUNA ROM DUMP TEST

THIS TEST WILL VERIFY THAT THE DATA PATH FROM THE T11 PROCESSOR TO THE UNIBUS INTERFACE IS INTACT AND ABLE TO TRANSFER DATA RELIABLY. THIS DATA PATH IS CRUCIAL FOR FURTHER TESTING BECAUSE IT IS NECESSARY FOR LOADING REPAIR-LEVEL DIAGNOSTICS INTO THE WCS.

THE TEST STRATEGY IS TO TRANSFER KNOWN DATA OVER THE DATA PATH AND TO VERIFY THE TRANSFERRED DATA.

THE DATA SOURCE FOR THE DUMP TEST IS THE ROM MICROCODE RESIDENT ON THE DEUNA PORT BOARD. A DUMP OF THE ROM WILL EXERCISE THE DATA PATH NEEDED FOR LOADING WCS AND THE ROM CONTENTS CAN BE VERIFIED. THE ROM MICROCODE WILL BE CHECKED BY VERIFYING THE CRC BYTES. THE CRC BYTES CHARACTERIZE THE DATA CONTENTS OF THE ROM AND ARE BURNED INTO THE ROM AT THE TIME OF MANUFACTURE. A FAILURE TO VERIFY THE CRC CALCULATION ON THE DUMPED ROM DATA DUMP WILL BE INTERPRETED AS AN ERROR IN THE DATA PATH.

TEST 11: WCS LOAD/DUMP TEST

THIS TEST WILL USE THE LOAD/DUMP PORT COMMAND TO VERIFY THE DATA PATHWAY TO/FROM THE WCS. PATTERNS WILL BE USED TO CHECK THE DATA PATHWAY FOR ALL SA0 AND SA1 ERRORS.

BECAUSE THE OPERATIONAL MICROCODE NEEDS THE LOWER 2K OF WCS ONLY THE TOP HALF OF WCS WILL BE LOADED WITH A DATA PATTERN THEN DUMPED BACK TO MEMORY FOR VERIFICATION. THIS PROCEDURE WILL BE REPEATED FOR ALL PATTERNS.

TEST 12: LOAD AND START FUNCTION TEST

THIS TEST WILL VERIFY THAT THE LOAD AND START MICROADDRESS PORT COMMAND IS OPERATIONAL.

THE PROCESS IS TO LOAD WCS WITH MICROCODE THAT WHEN STARTED WILL WRITE A PATTERN OF DATA TO THE LITE-BYTE FIELD OF PCSR1 REGISTER WHICH CAN BE READ FROM THE UNIBUS AND BE VERIFIED.

NOTE: THIS TEST USES MICROCODE MODULE 'A'

TEST 13: COMPREHENSIVE WCS MEMORY TEST

THIS TEST WILL EXHAUSTIVELY TEST THE WCS MEMORY.
CUSTOM MICROCODE MODULE B, MICROTEST #1 IS USED TO DO THE ACTUAL TESTING. MICROTEST #1 RUNS A SERIES OF MICROSUBTESTS TESTS ON THE WCS MEMORY CHECKING FOR BOTH ADDRESS AND DATA ERRORS. IF AN ERROR DOES OCCUR THE PORT CONTROL BLOCK WILL CONTAIN THE INFORMATION ABOUT THE ERROR.

PCBB+0: CONTAINS THE MICROSUBTEST THAT FAILED

PCBB+1: 0 = DATA ERROR, 1 = ADDRESS ERROR
PCBB+2: CONTAINS THE ADDRESS OF THE LOCATION
PCBB+4: CONTAINS THE DATA THAT WAS WRITTEN
PCBB+6: CONTAINS THE DATA THAT WAS READ

TEST 14: INTERRUPT VECTOR TEST

THIS TEST WILL VERIFY THAT THE INTERRUPT INTERFACE LOGIC OF THE DEUNA IS CAPABLE OF GENERATING AN INTERUPT VECTOR AND ARBITRATING FOR CONTROL OF THE UNIBUS.
THE DEUNA INTERRUPT ENABLE BIT WILL BE SET AND AN INTERRUPT WILL BE CAUSED BY ISSUING A NOP PORT COMMAND. AN INTERRUPT IS EXPECTED AT THE CORRECT VECTOR AND AT THE CORRECT PRIORITY.

TEST 15: PCSRO INTERRUPT BIT TEST

THIS TEST WILL VERIFY THAT EACH OF THE INTERRUPT BITS IN REGISTER PCSRO CAN CAUSE AN INTERRUPT.

EACH OF THE INTERRUPTS OF REGISTER PCSRO IS SET UNDER THE CONTROL OF THE T11 AND NOT DIRECTLY BY HARDWARE. THE T11 THEREFORE CAN INITIATE UNIBUS INTERRUPTS BY SETTING BITS IN REGISTER PCSRO.

THIS TEST USES MICROMODULE C, MICROTEST #1.
MICROCODE MODULE C IS LOADED IF NOT ALREADY DONE SO BY A PREVIOUS TEST.

THE DEUNA INTERRUPT VECTOR IS SETUP TO STORE THE CONTENTS OF PCSRO WHEN THE INTERRUPT OCCURS. PCBB+0 IS LOADED WITH THE INTERRUPT BIT THAT IS TO BE TESTED THEN PCSRO COMMAND BITS ARE LOADED WITH A 1 TO TELL THE T11 TO EXECUTE MICROTEST #1. WE WAIT FOR THE INTERRUPT TO OCCUR THEN SEE IF THE CONTENTS OF PCSRO AT THE TIME OF THE INTERRUPT CONTAINED THE CORRECT INTERRUPT BIT. THE TEST IS REPEATED FOR ALL THE INTERRUPT BITS.

TEST 16: TIMER TEST

THIS TEST WILL USE THE CUSTOM MICROCODE MODULE 'C' TO CHECK THE OPERATION OF THE TIMER.
THE TIMER IS ACCESSIBLE ONLY TO THE T11 PROCESSOR. THE HOST PROCESSOR CAN START THE TIMER ONLY WITH THE ASSISTANCE OF THE T11 PROCESSOR.

FOR THIS TEST THE MICROCODE WILL BE LOADED ONLY IF IT HAS NOT ALREADY BEEN DONE BY A PREVIOUS TEST.

WHEN THE MICROCODE IS STARTED THE T11 WILL START THE TIMER AND WILL SET 'DNI' WHEN THE TIMING INTERVAL HAS EXPIRED. THE INTERVAL IS 10 SECONDS.

ANY TIME FROM 8 TO 12 SECONDS IS AN ACCEPTABLE RANGE.

TEST 17: LINK MEMORY TEST

THIS TEST WILL EXHAUSTIVELY TEST THE LINK MEMORY.

THE LINK MEMORY OCCUPIES THE 16-32K ADDRESS SPACE OF THE T-11.
CUSTOM MICROCODE MODULE C MICROTEST #3 IS USED TO DO THE ACTUAL TESTING.
MICROTEST #3 RUNS A SERIES OF MICROSUBTESTS TESTS ON THE LINK MEMORY CHECKING
FOR BOTH ADDRESS AND DATA ERRORS. IF AN ERROR DOES OCCUR THE PORT CONTROL
BLOCK WILL CONTAIN THE INFORMATION ABOUT THE ERROR.

PCBB+0: CONTAINS THE MICROSUBTEST THAT FAILED
PCBB+1: 0 = DATA ERROR, 1 = ADDRESS ERROR.
PCBB+2: CONTAINS THE ADDRESS OF THE LOCATION
PCBB+4: CONTAINS THE DATA THAT WAS WRITTEN
PCBB+6: CONTAINS THE DATA THAT WAS READ

MICROSUBTEST #	DESCRIPTION
1	ACCESS TEST
2	ADDRESS SHIFT TEST
3	DATA LATCH TEST
4	ADDRESS BIT SHIFT #1
5	ADDRESS BIT SHIFT #2
6	MARCH TEST

TEST 18: DMA 'TO' ADDRESS TEST

THIS TEST WILL VERIFY THAT THE INTERNAL REGISTER 'DMATO' CAN BE READ
AND WRITTEN. THE T11 WILL BE USED TO WRITE AND READ THIS REGISTER.
THIS TEST REQUIRES THE USE OF CUSTOM MICROCODE MODULE C MICROTEST #4.
PCBB+0 WILL BE WRITTEN WITH THE DATA PATTERN TO TEST, THE T11 WILL
WRITE THIS PATTERN TO THE 'DMATO' REGISTER AND READ IT BACK AND PUT
THE DATA READ INTO PCBB+2. THE DATA AT PCBB+2 WILL BE VERIFIED.

TEST 19: DMA 'FROM' ADDRESS REGISTER TEST

THIS TEST CHECKS THE OPERATION OF THE REGISTER/COUNTER THAT CONTAINS
THE ADDRESS OF THE LINK MEMORY WORD TO BE MOVED TO THE HOST DURING
DMA OPERATIONS. THE REGISTER CAN BE WRITTEN BY THE T11 BUT IT CAN
NOT BE READ BACK FOR VERIFICATION, THEREFORE IT MUST BE CHECKED
INDIRECTLY.

THE METHOD USED IS TO LOAD MICROCODE MODULE C IF IT HAS NOT ALREADY
BEEN DONE. THE MICROTEST #5 LOADS EACH LOCATION OF LINK MEMORY WITH
ITS ADDRESS THEN IT TAKES THE CONTENTS OF PCBB+0 AND LOADS IT INTO
THE DMA 'FROM' ADDRESS REGISTER, THE 'TO' REGISTER IS LOADED WITH
THE ADDRESS OF PCBB+2, THE WORD COUNT IS LOADED FOR A ONE WORD TRANSFER
AND THE DMA ENGINE IS STARTED. THE HOST VERIFIES PCBB+2 = PCBB+0

TEST 20: DMA BLOCK TRANSFER TEST

THIS TEST WILL VERIFY THAT THE DMA ENGINE CAN TRANSFER A MAXIMUM SIZE DATA
BLOCK TO HOST MEMORY.

THIS TEST USES CUSTOM MICROCODE MODULE C, MICROTEST #6. THE MICROTEST
FILLS EACH LOCATION OF LINK MEMORY WITH ITS ADDRESS AND THEN SETS
UP A TRANSFER FROM LINK MEMORY TO THE ADDRESS POINTED TO BY PCBB+0.
THE TRANSFER SIZE IS 1776 WORDS. AFTER THE MICROTEST FINISHES THE
BUFFER IS CHECKED TO SEE IF IT CONTAINS THE INCREMENTING ADDRESS PATTERN.

TEST 21: TRANSMIT DONE TEST

THE TRANSMIT STATE MACHINE INFORMS THE PORT MODULE PROCESSOR OF A 'TRANSMIT DONE' CONDITION. IT DOES THIS BY GENERATING AN INTERRUPT WHENEVER IT FINISHES TRANSMITTING A DATAGRAM. SINCE THE 'TRANSMIT DONE' INTERRUPT IS A NECESSARY CONDITION OF EVERY DATAGRAM TRANSMISSION, THIS TEST WILL USE THE INTERRUPT TO INDICATE THAT THE TRANSMIT STATE MACHINE IS FUNCTIONING.

MICROCODE MODULE D MICROTEST #1 WILL BE USED FOR THIS TEST. IT SETS UP THE T-11 FOR AN INTERRUPT, STARTS A DATAGRAM LOOPBACK AND WAITS FOR A TRANSMIT INTERRUPT. THE T-11 WILL BE RELEASED FROM THE LOOP IF THE XMIT DONE INTERRUPT OCCURS. UPON RELEASE THE DNI BIT WILL BE SET IN PCSRO SIGNALING THAT THE TEST IS COMPLETE.

TEST 22: RECEIVER DONE TEST

THE LINK HARDWARE INCLUDES LOGIC TO TELL THE DEUNA PROCESSOR WHEN A LINK MEMORY BUFFER HAS BEEN FILLED AND DATA IS AVAILABLE FOR PROCESSING. THE HARDWARE INTERRUPTS THE DEUNA PROCESSOR. BECAUSE THE INTERRUPT HAPPENS WHEN A LINK MEMORY BUFFER IS FULL AND THE LINK MEMORY IS FILLED BY THE OPERATION OF THE RECEIVE STATE MACHINE, THE INTERRUPT CAN BE USED TO CHECK IF THE STATE MACHINE WORKS.

MICROCODE MODULE D MICROTEST #2 WILL BE USED FOR THIS TEST. IT SETS UP THE T-11 FOR AN INTERRUPT, STARTS A DATAGRAM LOOPBACK AND WAITS FOR A RECEIVER INTERRUPT. THE T-11 WILL BE RELEASED FROM THE LOOP IF THE INTERRUPT OCCURS. UPON RELEASE THE DNI BIT WILL BE SET IN PCSRO SIGNALING THAT THE TEST IS COMPLETE.

TEST 23: DATA BYTE FRAMING TEST

THIS TEST WILL CHECK THE LINK MODULE DATA PATH FOR BYTE DATA BOUNDARY CONDITIONS.

THE T-11 PROCESSOR WILL TRANSMIT DATA IN LOOPBACK MODE. THE DATA WILL BE ORGANIZED SUCH THAT DATA BOUNDARIES ARE CREATED BETWEEN ADJACENT BYTES IN THE DATA STREAM (I.E. 1111111100000000011...) THE T-11 PROCESSOR WILL VERIFY THE CONDITION OF THE DATA AFTER IT IS LOOPED BACK TO THE RECEIVER DATA BUFFER.

THIS TEST WILL USE MICROCODE MODULE 'D' MICROTEST #3. TESTING OF THE DATA FRAMING WILL BE DONE BY THE T-11 PROCESSOR. THE HOST PROCESSOR, MEANWHILE, WILL WAIT FOR A 'DNI' IN REGISTER PCSRO. IF 'DNI' APPEARS, THE HOST PROCESSOR WILL CHECK PCSR1 FOR AN ERROR CONDITION. IF AN ERROR CONDITION IS SET, ADDITIONAL ERROR INFORMATION WILL BE FOUND IN THE PCBB AS FOLLOWS:

PCBB+0: RECEIVER STATUS WORD
PCBB+2: DATA TRANSMITTED
PCBB+4: DATA RECEIVED
PCBB+6: WORD OFFSET INTO RECEIVER BUFFER OF BAD DATA

TEST 24: DATA WORD FRAMING TEST

THIS TEST WILL CHECK THE LINK MODULE DATA PATH FOR WORD DATA BOUNDARY CONDITIONS.

THE T-11 PROCESSOR WILL TRANSMIT DATA IN LOOPBACK MODE. THE DATA WILL BE ORGANIZED SUCH THAT DATA BOUNDARIES ARE CREATED BETWEEN ADJACENT WORDS IN THE DATA STREAM (I.E. 11111111111111110000000000000000011...) THE T-11 PROCESSOR WILL VERIFY THE CONDITION OF THE DATA AFTER IT IS LOOPED BACK TO THE RECEIVER DATA BUFFER.

THIS TEST WILL USE MICROCODE MODULE 'D' MICROTEST #4. TESTING OF THE DATA FRAMING WILL BE DONE BY THE T-11 PROCESSOR. THE HOST PROCESSOR, MEANWHILE, WILL WAIT FOR A 'DNI' IN REGISTER PCSRO. IF 'DNI' APPEARS, THE HOST PROCESSOR WILL CHECK PCSR1 FOR AN ERROR CONDITION. IF AN ERROR CONDITION IS SET, ADDITIONAL ERROR INFORMATION WILL BE FOUND IN THE PCBB AS FOLLOWS:

PCBB+0: RECEIVER STATUS WORD
PCBB+2: DATA TRANSMITTED
PCBB+4: DATA RECEIVED
PCBB+6: WORD OFFSET INTO RECEIVER BUFFER OF BAD DATA

TEST 25: DATA PATH PATTERN TEST

THIS TEST WILL CHECK THE LINK MODULE DATA PATH FOR ALL 'STUCK AT 0' AND 'STUCK AT 1' ERRORS.

THE T-11 PROCESSOR WILL TRANSMIT DATAGRAMS OF MAXIMUM LENGTH IN LOOPBACK MODE. THIS PATTERN LOOPBACK PROCEDURE WILL BE USED FOR ALL PATTERNS OF UP TO WORD WIDTH.

THIS TEST USES MICROMODULE 'D' MICROTEST #5 TO DO THE TESTING. THE HOST PROCESSOR WILL PASS A DATA PATTERN TO THE T-11 PROCESSOR THROUGH THE PCBB. THE T-11 WILL FILL A XMIT BUFFER WITH THE DATA PATTERN AND TRANSMIT THE DATAGRAM OVER THE LOOPBACK. THE T-11 PROCESSOR WILL VERIFY THE PATTERN IN THE RECEIVER BUFFER. IF THE T-11 FINDS AN ERROR, IT WILL WRITE THE FAILING PATTERN TO THE PCBB ALONG WITH THE OFFSET INTO THE RECEIVER BUFFER AT WHICH THE PATTERN WAS FOUND. IT WILL INFORM THE HOST OF THE ERROR BY SETTING PCSR1 TO AN ERROR CONDITION. THE PCBB IS FORMATTED AS FOLLOWS:

PCBB+0: DATA PATTERN
PCBB+2: RECEIVER STATUS WORD
PCBB+4: BAD DATA PATTERN
PCBB+6: OFFSET INTO RECEIVER BUFFER WHERE BAD DATA WAS FOUND

TEST 26: STATUS MUX VERIFICATION TEST

THE LINK WRITES STATUS IN LINK MEMORY AFTER EACH TRANSMIT ATTEMPT. THE STATUS GIVES INFORMATION ABOUT THE ATTEMPTED OPERATION. THE STATUS INFORMATION IS WRITTEN INTO THE FIRST TWO LOCATIONS OF THE TRANSMIT BUFFER. THIS INFORMATION IS ACCESSIBLE TO THE T-11 BY SIMPLY READING IT FROM LINK MEMORY.

THIS TEST WILL VERIFY THAT THE STATUS INFORMATION IS WRITTEN INTO THE FIRST LOCATION OF THE TRANSMIT BUFFER. THE TEST WILL ALSO CHECK THE SECOND WORD OF THE TRANSMIT BUFFER.

THIS TEST WILL USE MICROMODULE 'D' MICROTEST #6.
WHEN THE TEST IS STARTED, THE T-11 PROCESSOR WILL SET UP THE LINK FOR LOOPBACK OF A DATA PATTERN. A BACKGRAOUND PATTERN WILL BE WRITTEN INTO THE FIRST WORD OF THE TRANSMIT BUFFER. THIS WORD SHOULD BE OVER-WRITTEN BY THE STATUS WHEN

THE BUFFER IS TRANSMITTED. THE SECOND WORD OF THE TRANSMIT BUFFER CAN NOT BE WRITTEN WITH A BACKGROUND BECAUSE IT MUST DESIGNATE THE TRANSMIT BYTE COUNT.

WHEN THE DATAGRAM HAS BEEN LOOPED BACK, THE T-11 PROCESSOR WILL PASS THE FIRST TWO WORDS OF THE TRANSMIT BUFFER TO THE HOST THRU THE PCBB+0 AND PCBB+2.

PCBB+0: FIRST WORD OF TRANSMIT BUFFER
PCBB+2: SECOND WORD OF TRANSMIT BUFFER

THE CORRECT STATUS SHOULD BE:

```
        TRANSMIT STATUS WORD 0 BITS 15,09:00 SHOULD BE ALL 0 AND
                               BIT 13 SHOULD BE A 1

        TRANSMIT STATUS WORD 1 BITS 15:13 SHOULD ALL BE 0
```

TEST 27: LINK BYTE COUNTER TEST

BYTE COUNTERS ARE INVOLVED BOTH WITH THE LINK TRANSMIT FUNCTION AND THE LINK RECEIVE FUNCTION. WHEN THE T-11 PREPARES A TRANSMIT BUFFER FOR TRANSMISSION OF A DATAGRAM, IT WRITES THE INTENDED BYTE COUNT IN THE SECOND WORD OF THE TRANSMIT BUFFER. WHEN TRANSMISSION OF THE TRANSMIT BUFFER BEGINS, THE BYTE COUNT VALUE IS USED TO LOAD THE TRANSMIT BYTE COUNTER. THIS COUNTER IS DECREMENTED BY THE TRANSMIT STATE MACHINE AS THE DATAGRAM IS TRANSMITTED. THE DATAGRAM TRANSMISSION WILL CONTINUE UNTIL THE BYTE COUNTER IS DECREMENTED TO ZERO.

THE RECEIVER ALSO HAS A BYTE COUNTER. THIS COUNTER IS CLEARED AT THE START OF A DATAGRAM RECEPTION AND IS INCREMENTED BY THE RECEIVE STATE MACHINE AS THE DATAGRAM IS RECEIVED. THE VALUE IN THIS COUNTER IS WRITTEN INTO WORD TWO OF THE RECEIVE BUFFER AT THE END OF RECEPTION.

THIS TEST WILL USE MICROMODULE 'D' MICROTEST #7.
THIS TEST WILL VERIFY THE BYTE COUNT LOGIC BY LOOPING MESSAGES AND VERIFYING THAT THE BYTE COUNT APPEARING IN THE RECEIVE BUFFER CORRESPONDS TO THE BYTE COUNT THAT WAS WRITTEN TO THE TRANSMIT BYTE COUNTER. THE TEST WILL ALSO VERIFY THAT THE ACTUAL NUMBER OF BYTES TRANSFERRED TO THE RECEIVE BUFFER CORRESPONDS TO THE INTENDED BYTE COUNT.

THE TRANSMIT BYTE COUNT IS PASSED TO THE T-11 VIA THE PCBB+0. AFTER THE DATAGRAM LOOPBACK THE RECEIVE BYTE COUNT IS PLACED INTO PCBB+2 BY THE T-11 PROCESSOR. PCBB+4 IS LOADED BY THE T-11 PROCESSOR WITH THE ACTUAL NUMBER OF BYTES THAT WERE TRANSFERRED TO THE RECEIVER BUFFER.

PCBB+0: TRANSMIT BYTE COUNT
PCBB+2: RECEIVE BYTE COUNT
PCBB+4: ACTUAL NUMBER OF BYTES RECEIVED

TEST 28: ODD BYTE TEST

THIS TEST WILL VERIFY THAT THE LINK CAN TRANSMIT AND RECEIVE DATAGRAMS HAVING ONLY ODD BYTE COUNTS.

THIS TEST IS IDENTICAL TO THE PREVIOUS BYTE COUNTER TEST WITH THE ONLY

EXCEPTION THAT IT PASSES ONLY ODD BYTE COUNTS TO THE MICROCODE. IT ALSO USES MICROMODULE 'D' MICROTEST #7

TEST 29: LINK MAXIMUM BYTE COUNTER TEST

THE RECEIVE BYTE COUNTER IS A 12 BIT BINARY COUNTER THAT COUNTS THE NUMBER OF BYTES THAT WERE RECEIVED DURING A DATAGRAM TRANSMISSION. THE BYTE COUNTER IS INCREMENTED AS EACH BYTE IS RECEIVED. THE RECEIVE BYTE COUNTER HAS LOGIC THAT DISABLES THE COUNTER IF THE MAXIMUM VALUE IS REACHED AND PREVENTS THE COUNTER FROM ROLLING OVER TO ZERO.

THIS TEST WILL CHECK THAT THE COUNTER 'TOPS OUT' AT THE MAXIMUM COUNTER VALUE. IT DO THIS MICROMODULE 'D' MICROTEST #9 IS USED. IT WILL TRANSMIT A DATAGRAM OF MAXIMUM COUNTER LENGTH OVER THE LOOPBACK. THE LINK CRC HARDWARE WILL BE ALLOCATED TO THE TRANSMIT SIDE SO THAT CRC BYTES WILL APPENDED TO THE DATAGRAM. THE LENGTH OF THE DATAGRAM WILL THEREFORE EXCEED THE LENGTH OF THE RECEIVE BYTE COUNTER. THE RECEIVE COUNTER WILL BE CHECKED TO INSURE THAT THE COUNTER HAS REMAINED AT THE MAXIMUM VALUE, IF NOT AN ERROR IS PASSED TO THE HOST.

TEST 30: FIFO TEST

THERE ARE TWO FIFO'S USED IN THE DEUNA TO KEEP TRACK OF RECEIVER BUFFERS. THE FIRST IS CALLED THE 'RECEIVER BUFFER AVAILABLE FIFO' AND THE SECOND IS CALLED THE 'RECEIVER BUFFER DONE FIFO'.

THE T11 LOADS THE RECEIVER BUFFER AVAILABLE FIFO WITH A LIST OF UNUSED 1K BUFFERS IN LINK MEMORY. WHEN THE DEUNA SENSES THAT A PACKET IS COMING IN IT PULLS AN AVAILABLE BUFFER ADDRESS FROM THE OUTPUT OF THE RECEIVER BUFFER AVAILABLE FIFO AND USES IT TO ADDRESS LINK MEMORY FOR THE STORAGE OF THE RECEIVED DATA. AFTER THE DATA HAS BEEN LOADED THE RECEIVER STATE MACHINE PUTS THE USED BUFFER ADDRESS INTO THE RECEIVER BUFFER DONE FIFO WHERE AN INTERRUPT IS GENERATED TO THE T11 WHEN IT BUBBLES TO THE TOP OF THE FIFO.

THESE FIFO'S ARE 64 DEEP BY 4 BITS WIDE. THE OPERATIONAL MICROCODE ONLY FILLS THE FIFO TO A MAXIMUM OF 16. THE 4 BIT WIDTH REPRESENTS BITS 14-11 OF THE LINK MEMORY ADDRESS. THESE BITS ALLOW THE ADDRESSING OF A 1K BUFFER IN LINK MEMORY.

THIS TEST WILL VERIFY THAT THE RECEIVE BUFFER AVAILABLE FIFO AND THE RECEIVER BUFFER DONE FIFO OPERATE CORRECTLY. THIS WILL BE DONE BY LOADING THE RECEIVER BUFFER AVAILABLE FIFO WITH A 1K BUFFER ADDRESS THEN A PACKET WILL BE TRANSMITTED IN LOOPBACK MODE. AFTER THE RECEIVER INTERRUPT OCCURS THE RECEIVER BUFFER DONE FIFO IS READ AND THE ADDRESS IS COMPARED WITH WHAT WAS GIVEN THE RECEIVER BUFFER AVAILABLE FIFO. THEY SHOULD BE THE SAME IF EVERYTHING IS WORKING CORRECTLY. THE OPERATION IS PERFORMED WITH THE TRANSMITTER BUFFER SET TO 0 AND WILL BE REPEATED WITH RECEIVER BUFFERS 1-15.
THIS TEST WILL USE MICROMODULE 'D' MICROTEST #10.

PARAMETERS PASSED TO THE MICROCODE WILL BE FORMATTED IN THE PCBB AS FOLLOWS:

PCBB+0: RECEIVE BUFFER ADDRESS
PCBB+2: RECEIVE BUFFER COMPLETED ADDRESS

TEST 31: RECEIVER LINK MEMORY ADDRESS TEST

THIS TEST WILL VERIFY THAT BUFFERS 1-15 OF LINK MEMORY CAN BE ADDRESSED CORRECTLY BY THE RECEIVER. THIS WILL BE DONE BY DIRECTING THE MICROCODE TO TRANSMIT A DATA PATTERN FROM BUFFER 0 AND TO RECEIVE THE DATA IN BUFFER X WHERE X = 1-15. THEN A CHECK WILL BE MADE TO SEE IF THE PATTERN NOT ONLY ARRIVED IN THE CORRECT RECEIVER BUFFER BUT THAT THE PATTERN DOES NOT SHOW UP ANYWHERE ELSE IN LINK MEMORY EXCEPT WHERE IT WAS SUPPOSE TO.

THIS TEST WILL USE MICROMODULE 'D' MICROTEST #11. THIS MICROTEST ACCEPTS 2 PARAMETERS: THE TRANSMIT BUFFER AND THE RECEIVER BUFFER. IT WILL SET UP A DATA PATTERN IN THE TRANSMIT BUFFER AND TELL THE LINK TO TRANSMIT, IN LOOPBACK MODE, FROM THE TRANSMIT BUFFER GIVEN TO THE RECEIVER BUFFER GIVEN. AFTER THE RECEIVER INTERRUPT, THE DATA IS CHECKED IN THE EXPECTED RECEIVER BUFFER FOR THE CORRECT DATA PATTERN. THEN ALL OF LINK MEMORY (EXCEPT FOR THE TRANSMIT BUFFER) IS CHECKED TO SEE IF THE PATTERN ENDS UP ELSEWHERE. IF AN ERROR IS FOUND THE MICROCODE PASSES THE ADDRESS OF LINK MEMORY WHERE THE ERROR WAS FOUND, THE DATA THAT WAS FOUND THERE ALONG WITH THE DATA THAT SHOULD HAVE BEEN THERE.

THE PARAMETERS FOR THE MICROCODE ARE FORMATED IN THE PCBB AS FOLLOWS:

PCBB+0: RECEIVER BUFFER ADDRESS
PCBB+2: TRANSMIT BUFFER ADDRESS
PCBB+4: LINK MEMORY ADDRESS (IF ERROR)
PCBB+6: GOOD DATA PATTERN (IF ERROR)
PCBB+10: BAD DATA PATTERN (IF ERROR)

TEST 32: TRANSMITTER LINK MEMORY ADDRESS TEST

THIS TEST WILL VERIFY THAT BUFFERS 1-15 CAN BE CORRECTLY ADDRESSED BY THE TRANSMITTER. THIS TEST IS IDENTICAL TO THE RECEIVER ADDRESS TEST IN THAT IT USES THE SAME MICROCODE (MICROMODULE 'D' MICROTEST #11) EXCEPT IT FIXES THE RECEIVER BUFFER AT 0 AND VARIES THE TRANSMIT BUFFER FROM 1-15.

TEST 33: LINK MEMORY ARBITRATION TEST

THE LINK MEMORY CAN BE ACCESSED BY FOUR PROCESSES; THE T-11 PROCESSOR, THE DMA ENGINE, THE RECEIVE STATE MACHINE AND THE TRANSMIT STATE MACHINE. THE PORT MODULE HAS ARBITRATION CIRCUITRY TO MANAGE LINK MEMORY ACCESSES. THIS CIRCUITRY PREVENTS CONFLICKS BETWEEN PROCESSES AND ASSURES THAT HIGHER PRIORITY PROCESSES GET PRECEDENCE.

THIS TEST WILL VERIFY THE ABLILITY OF THE LINK MEMORY ARITRATOR TO HANDLE SIMULTANEOUS REQUESTS BY FOUR PROCESSES. EACH OF THESE PROCESSES WILL INVOLVE TASKS THAT REQUIRE HEAVY ACCESSES OF LINK MEMORY. DATA WILL BE MOVED INTO OR OUT OF LINK MEMORY BY EACH. WHEN THAT TASKS ARE FINISHED THE DATA WILL BE VERIFIED.

THE FOUR PROCESSES ARE:

1-TRANSMIT STATE MACHINE

WILL TRANSMIT A DATAGRAM OF MAXIMUM DATA LENGTH IN LOOPBACK

```
                MODE. THE DATA FIELD WILL CONTAIN A BIT PATTERN STRING OF
                TWO 1'S FOLLOWED BY TWO 0'S I.E. 31463 (OCTAL).

        2-RECEIVE STATE MACHINE

                WILL RECEIVE A DATAGRAM OF MAXIMUM DATA LENGTH OVER THE
                LOOPBACK. THE RECEIVE DATA BUFFER WILL BE FILLED WITH 0'S
                PRIOR TO THE RECEPTION.

        3-T-11 MICROPROCESSOR DMA

                A 1K BUFFER IN LINK MEMORY WILL BE FILLED WITH AN ALL 1'S DATA
                PATTERN PRIOR TO THE OPERATION THEN ALTERNATING 1'S AND 0'S
                DATA PATTERN WILL BE WRITTEN.

        4-DMA ENGINE

                WILL TRANSFER A 1K BLOCK OF DATA FROM LINK MEMORY TO UNIBUS
                MEMORY. THE DATA IN LINK MEMORY WILL A BIT PATTERN STRING
                OF FOUR 1'S FOLLOWED BY A STRING OF FOUR 0'S. THE BUFFER
                IN UNIBUS MEMORY WILL BE CLEARED PRIOR TO THE OPERATION.

THE FOUR PROCESSES WILL WORK OUT OF FOUR SEPARATE AREAS OF LINK MEMORY.

BASE OF LINK MEMORY-->  +-------------------------+
                        ! RECEIVE BUFFER          !
        BASE + 4000---> +-------------------------+
                        ! TRANSMIT BUFFER         !
        BASE + 10000--> +-------------------------+
                        ! DMA ENGINE BUFFER       !
        BASE + 14000--> +-------------------------+
                        ! T-11 PROCESSOR BUFFER   !
                        +-------------------------+

THIS WILL ALLOW THE ARITRATION CIRCUITRY TO BE TESTED AND YET ALLOWS THE DATA
TO BE VERIFIED EASILY AND ASSOCIATED WITH A SINGLE PROCESS.

A DATAGRAM WILL BE LOOPED BACK FROM THE TRANSMIT BUFFER TO THE RECEIVE BUFFER.
AS THE DATAGRAM IS BEING TRANSFERRED, THE T-11 PROCESSOR WILL FILL IT'S BUFFER
AND THE DMA ENGINE WILL TRANSFER IT'S BUFFER FROM LINK MEMORY TO UNIBUS MEMORY.

WHEN THE RECEIVE STATE MACHINE IS DONE, THE T-11 WILL VERIFY THE DATA IN THE
RECEIVE BUFFER. IF AN ERROR IS FOUND PCSR1 WILL BE SET TO INDICATE AN ERROR.

THE HOST WILL WAIT FOR THE MICROCODE TO FINISH AND WHEN DONE, IT WILL VERIFY
THE DATA TRANSFERRED BY THE DMA ENGINE TO UNIBUS MEMORY.

THE FIRST 3 WORDS OF THE PCBB ARE USED FOR ERROR INFORMATION. THE REST
WILL BE THE UNIBUS ADDRESS THAT THE DMA ENGINE WILL TRANSFER TO.

                +-------------------------+
 PCBB+0:        !    EXPECTED PATTERN     !
                +-------------------------+
 PCBB+2:        !     ACTUAL PATTERN      !
                +-------------------------+
 PCBB+4:        !   LINK MEMORY ADDRESS   !
```

```
               +------------------------+
PCBB+6:        !   DATA TRANSFERRED     !
   !           !         FROM           !
   !           !      LINK MEMORY       !
   !           !        BY THE          !
   !           !      DMA ENGINE        !
   V           !                        !
PCBB+4000:     +------------------------+
```

TEST 34: STATION ADDRESS PATTERN TEST

WITHOUT EITHER THE PROMISCUIOUS MODE OR THE MULTICAST MODE ENABLED, THE LINK LOGIC WILL RECOGNIZE DATAGRAM ADDRESSES ONLY IF THE ADDRESS IS CONTAINED IN THE STATION ADDRESS RAM.

WHEN A DATAGRAM ARRIVES, THE LINK LOGIC COMPARES THE DATAGRAM DESTINATION ADDRESS FIELD TO THE 12 ADDRESSES WRITTEN IN THE STATION ADDRESS RAM. IF THE INCOMING ADDRESS MATCHES ONE OF THESE, THEN THE DATAGRAM WILL BE ACCEPTED BY THE LINK. THE 'MATCH' BIT IS SET IN THE TRANSMIT BUFFER AND THE RECEIVING PROCESS BEGINS.

THIS TEST WILL VERIFY THAT THE LINK CAN RECOGNIZE A DATAGRAM WHEN THE DESTINATION ADDRESS OF THE DATAGRAM MATCHES ONE OF THE ADDRESSES STORED IN THE STATION ADDRESS RAM.

THIS TEST WILL USE MICROMODULE 'E' MICROTEST #1.
PATTERNS WILL BE USED FOR ADDRESSES IN CHECKING THE STATION ADDRESS LOGIC. THE PATTERNS WILL BE SUPPLIED TO THE T-11 THROUGH THE PCBB. THE MICROCODE WILL BE RESTARTED FOR EACH DIFFERENT PATTERN TO BE TESTED. UPON START-UP, THE T-11 PROCESSOR WILL PICK UP THE CURRENT PATTERN/ADDRESS, LOAD THE SAME PATTERN INTO ALL 12 LOCATIONS OF THE STATION ADDRESS RAM, FORMAT THE TRANSMIT BUFFER AND LOGIC FOR A LOOPBACK, PRESET PCSR1 TO AN ERROR CONDITION, START THE LINK AND WAIT FOR THE MATCH BIT IN THE TRANSMIT BUFFER. IF THE MATCH BIT SETS THE PCSR1 ERROR CONDITION IS CLEARED AND THE T-11 WAITS FOR BOTH THE TRANSMITTER AND RECEIVER INTERRUPTS BEFORE IT SETS 'DNI' TO INDICATE THE TEST WAS SUCCESSFUL

THE PCBB WILL BE USED TO PASS THE 48 BIT STATION ADDRESS PATTERN:

```
              +-------------------------------+
PCBB+0:       !     STATION ADDRESS LOW       !
              +-------------------------------+
PCBB+2:       !    STATION ADDRESS MIDDLE     !
              +-------------------------------+
PCBB+4:       !     STATION ADDRESS HIGH      !
              +-------------------------------+
```

THE FOLLOWING PATTERNS WILL BE USED:
- -ALTERNATING 1'S AND 0'S
- -ALTERNATING 0'S AND 1'S
- -PAIR OF 0'S FOLLOWED BY PAIR OF 1'S
- -FOUR 0'S FOLLOWED BY FOUR 1'S
- -EIGHT 0'S FOLLOWED BY EIGHT 1'S
- -SIXTEEN 1'S FOLLOWED BY SIXTEEN 0'S FOLLOWED BY 16 1'S

-TWENTYFOUR 1'S FOLLOWED BY TWENTYFOUR 0'S

TEST 35: STATION ADDRESS REJECTION TEST

THIS TEST WILL VERIFY THAT THE STATION ADDRESS DETECTION LOGIC DOES NOT RECOGNIZE A DATAGRAM WHEN THE DATAGRAM ADDRESS IS NOT CONTAINED IN THE STATION ADDRESS RAM.

THE MICROCODE WILL FILL THE STATION ADDRESS RAM WITH 0'S. THE DESTINATION FIELD OF THE TRANSMIT BUFFER IS FILLED WITH 1'S . A TRANSMISSION IS STARTED IN LOOPBACK MODE AND THE T-11 WILL WAIT FOR A RECEIVER INTERRUPT. OF COURSE, THE RECEIVER INTERRUPT SHOULD NEVER HAPPEN BECAUSE THE STATION ADDRESS LOGIC SHOULD NOT GET A SUCCESSFUL COMPARISON BETWEEN 0'S IN THE DESTINATION ADDRESS OF THE INCOMING DATAGRAM AND THE 1'S IN THE STATION ADDRESS RAM. THE T-11 WILL BE PUT INTO A LOOP WAITING FOR A RECEIVER INTERRUPT AND THE DEUNA TIMER IS STARTED. IF THE LOOP IS BROKEN BY THE RECEIVER INTERRUPT AN ERROR WILL BE PRESENTED IN PCSR1 BY THE MICROCODE. IF THE LOOP IS BROKEN BY THE TIMER THEN THE TEST WAS SUCCESSFUL.

TEST 36: STATION ADDRESS RAM POSITION TEST

THE STATION ADDRESS RAM CAN HOLD UP TO 12 STATION ADDRESSES. WHEN A DATAGRAM IS RECEIVED THE STATION ADDRESS COMPARISON LOGIC DOES A BIT-WISE COMPARISON OF ALL 12 RAM STATION ADDRESS WITH THE INCOMING DATAGRAM STATION ADDRESS.

THIS TEST WILL VERIFY THAT THE DEUNA CAN RECOGNIZE A STATION ADDRESS REGARDLESS OF THE LOCATION OF THE ADDRESS IN THE STATION ADDRESS RAM.

THIS TEST WILL USE MICROMODULE 'E' MICROTEST #4. THE MICROCODE WILL WRITE A STATION ADDRESSES OF ALL 1'S INTO A SINGLE LOCATION OF THE STATION ADDRESS RAM. THE OTHER ELEVEN LOCATION WILL BE LOADED WITH 0'S. A DATAGRAM WITH AN ALL 1'S DESTINATION ADDRESS WILL BE TRANSMITTED IN LOOPBACK MODE. THE TEST WILL VERIFY THAT THE DATAGRAM IS RECEIVED. THE TEST WILL BE REPEATED FOR ALL TWELVE LOCATIONS OF THE STATION ADDRESS RAM.

THE MICROTEST WILL BE REPEATED FOR EACH OF THE 12 TEST ITERATIONS. THE PCBB WILL BE USED TO PASS TO THE MICROCODE WHICH POSITION IS TO BE LOADED WITH 1'S. WHEN THE STATION ADDRESS IS LOADED, THE STATION ADDRESSES MUST BE ROTATED ORTHOGONALLY, I.E. BIT 0 OF ALL STATION ADDRESSES LOADED TOGETHER, THEN BIT 1, THEN BIT 2 ETC. THIS MAKES IT DIFFICULT TO DESCRIBE THE POSITION OF ANY SINGLE STATION ADDRESS IN TERMS OF AN OFFSET FROM THE RAM BASE ADDRESS.

THE PCBB IS FORMATTED AS FOLLOWS:

```
                 +-------------------------+
PCBB+0:          !  RAM ADDRESS POSTION    !
                 +-------------------------+
```

TEST 37: MULTICAST ADDRESS TEST

MULTICAST ADDRESSING PERMITS THE DEUNA TO RESPOND TO MESSAGES AIMED AT LOGICALLY RELATED DEVICES ON THE NETWORK. THE MSB OF THE DESTINATION ADDRESS

OF THESE MESSAGES IS A 1. THIS BIT IS DETECTED BY THE ADDRESS RECOGNITION LOGIC.

THIS TEST WILL VERIFY THAT THE DEUNA CAN RECOGNIZE AND ACCEPT MESSAGES WITH THE MULTICAST BIT DESIGNATION.

THIS TEST WILL USE MICROMODULE 'E' MICROTEST #4.
THE MICROCODE WILL PREPARE A DATAGRAM WITH THE DESTINATION ADDRESS HAVING THE MULTICAST BIT SET. THE DEUNA WILL BE SETUP IN LOOPBACK MODE WITH 'ENABLE ALL MULTICAST'. THE DATAGRAM WILL BE TRANSMITTED AND THE T-11 WILL BE PUT IN A LOOP WAITING FOR A RECEIVER INTERRUPT. THE TIMER WILL INTERRUPT THE LOOP IF THE RECEIVER INTERRUPT DOES NOT OCCUR. IF THIS HAPPENS, PCSR1 WILL INDICATE AN ERROR, OTHERWISE WHEN THE RECEIVER INTERRUPT OCCURS IT WILL BREAK THE LOOP AND PCSR1 WILL INDICATE A SUCCESSFULL COMPLETION OF THE TEST.

TEST 38: CRC DATA PATTERN TEST

THE LINK MODULE HAS HARDWARE TO GUARANTEE THAT DATAGRAMS HAVE NOT BEEN CORRUPTED DURING TRANSMISSION AND RECEPTION. THE HARDWARE GENERATES A CRC FOR DATAGRAMS TRANSMITTED AND VERIFIES THE CRC FOR DATAGRAMS RECEIVED.
THE CRC IS A 32 BIT NUMBER GENERATED BY DIVIDING THE DATAGRAM BIT STREAM BY A CRC POLYNOMIAL. THE DIVISION RESULTS IN A UNIQUE NUMBER THAT CAN ONLY BE REPRODUCED IN CRC CALCULATIONS IF THE BIT STREAM EXACTLY MATCHES THE ORIGINAL.
THE CRC IS CALCULATED DURING DATAGRAM TRANSMISSION AND IS APPENDED TO THE PACKET. THE CRC IS TRANSMITTED AS PART OF THE PACKET. THE CRC IS AGAIN CALCULATED WHEN THE DATAGRAM IS RECEIVED AND THE CALCULATED IS COMPARED TO THE CRC TRANSMITTED. IF THE DATAGRAM HAS BEEN FAITHFULLY TRANSMITTED, THE CRC'S SHOULD MATCH EXACTLY.

THE DEUNA CALCULATES THE CRC WITH DEDICATED CRC LOGIC. THE LOGIC IS EITHER DEDICATED TO THE CALCULATION OF THE OUTGOING DATAGRAM OR THE CALCULATION OF THE INCOMING DATAGRAM, BUT NOT BOTH.

THIS TEST WILL VERIFY THE OPERATION OF THE CRC CALCULATION CIRCUITRY.
MICROMODULE 'F' MICROTEST #1 WILL BE USED.
THE MICROCODE WILL TRANSMIT DATAGRAMS IN LOOPBACK MODE. THE CRC HARDWARE WILL BE DEDICATED TO THE TRANSMITTER. WHEN THE DATAGRAM IS RECEIVED THE T-11 WILL CALCULATE A CRC ON THE DATA RECEIVED (INCLUDING THE TRANSMITTED CRC).
THE RESULT OF THIS CALCULATION WILL BE A 32 BIT CONSTANT. THIS CONSTANT IS THEN COMPARED TO WHAT WAS EXPECTED AND IF THEY DO NOT MATCH. AN ERROR IS PLACED IN PCSR1.

PATTERNS WILL BE PASSED TO THE MICROCODE THROUGH THE PCBB. THE MICROCODE WILL FILL THE TRANSMIT BUFFER WITH THIS PATTERN BEFORE EACH TRANSMISSION TAKES PLACE.

THE PCBB WILL BE FORMATTED AS FOLLOWS:

```
                +--------------+
PCBB+0:         ! DATA PATTERN !
                +--------------+
```

TEST 39: CRC ERROR TEST

THIS TEST WILL VERIFY THAT THE LINK CRC CIRCUITRY CAN DETECT A BAD CRC.

MICROMODULE 'F' MICROTEST #2 WILL BE USED. THE MICROCODE WILL TRANSMIT DATAGRAMS IN LOOPBACK MODE. EACH DATAGRAM WILL HAVE AN ERRONEOUS CRC APPENDED TO THE DATA FIELD. THE DEUNA CRC LOGIC WILL BE SETUP SUCH THAT THE CRC LOGIC WILL BE DEDICATED TO THE RECEIVER. THIS IS EXPECTED TO CAUSE A CRC ERROR.

THE DATA FIELDS OF EACH DATAGRAM WILL CONSIST OF PATTERNS. THE PATTERNS WILL BE PASSED TO THE MICROCODE VIA THE PCBB.

AFTER THE RECEIVER INTERRUPT THE MICROCODE WILL PASS THE RECEIVER STATUS WORD 0 BACK VIA PCBB+2. THE CRC BIT IN THIS WORD IS CHECKED TO SEE IF IT IS SET.

THE PCBB IS FORMATTED AS FOLLOWS:

```
                 +-------------------------+
PCBB+0:          !      DATA PATTERN       !
                 +-------------------------+
PCBB+2:          ! RECEIVER STATUS WORD    !
                 +-------------------------+
```

TEST 40: CRC PATTERN LENGTH TEST

THIS TEST WILL VERIFY THAT THE RECEIVE CRC HARDWARE CAN CALCULATE CRC FOR DATAGRAMS OF VARYING LENGTHS.

DATAGRAMS WILL BE TRANSMITTED FOR THE TRANSMIT BUFFER TO THE RECEIVE BUFFER IN LOOPBACK MODE. THE TRANSMIT CRC WILL BE DISABLED WHICH WILL ASSIGN THE CRC LOGIC TO CALCULATION OF INCOMING DATAGRAMS. THE CRC FOR TRANSMIT DATAGRAMS WILL BE CALCULATED BY THE MICROCODE. IT IS EXPECTED THAT THE CRC LOGIC WILL VERIFY THE CRC APPENDED TO THE DATAGRAM AS IT IS BEING RECEIVED.

PATTERNS WILL BE USED TO FILL THE DATAGRAM DATA FIELD. THE PATTERNS WILL BE PASSED TO THE MICROCODE THROUGH THE PCBB ALONG WITH THE BYTE COUNT TO BE USED.

AFTER THE RECEPTION OF THE DATAGRAM THE RECEIVER STATUS WORD WILL BE PASSED BACK VIA THE PCBB SO IT CAN BE CHECKED

THE PCBB IS FORMATTED AS FOLLOWS:

```
                 +-------------------------+
PCBB+0:          !      DATA PATTERN       !
                 +-------------------------+
PCBB+2:          !       BYTE COUNT        !
                 +-------------------------+
PCBB+4:          !  RECEIVE STATUS WORD    !
                 +-------------------------+
```

TEST 41: RECEIVER BUFFER RECOVERY - RUNT TEST

THIS TEST WILL CHECK THE ABILITY OF THE RECEIVE STATE MACHINE TO REJECT A DATAGRAM OF LESS THAN 64 BYTES AND TO RECOVER THE RECEIVER BUFFER.

THIS TEST WILL USE MICROMODULE 'F' MICROTEST #4.
EACH TRIAL WILL CONSIST OF TWO DATAGRAM TRANSMISSIONS IN LOOPBACK MODE. EACH TRANSMISSION WILL LOOPBACK A DATAGRAM FILLED WITH UNIQUE DATA. THE FIRST DATAGRAM WILL BE A RUNT OF LESS THAN 64 BYTES. THE SECOND WILL BE A DATAGRAM OF LEGAL SIZE.

EACH TRIAL WILL START WITH THE LINK BUFFER POINTER RESET TO THE FIRST LINK BUFFER. THE RUNT WILL BE TRANSMITTED, THEN THE VALID DATAGRAM. IF THE BUFFER RECOVERY IS WORKING CORRECTLY, THE SECOND DATAGRAM IS EXPECTED TO BE WRITTEN INTO THE SAME LINK MEMORY BUFFER AS WAS THE RUNT.

THIS TEST WILL BE REPEATED WITH VARIOUS RUNT PACKET SIZES.

THE BYTE COUNT FOR THE RUNT PACKET TRANSMISSION WILL BE PASSED VIA THE PCBB. AFTER THE TWO TRANSMISSIONS, THE MICROCODE WILL PASS BACK THE CONTENTS OF THE BUFFER DONE FIFO, AND THE CONTENTS OF THE FIRST DATA WORD OF THE RECEIVER BUFFER.

THE PCBB WILL BE FORMATTED AS FOLLOWS:

```
                 +--------------------------------+
PCBB+0:          !        RUNT BYTE COUNT         !
                 +--------------------------------+
PCBB+2:          !   BUFFER DONE FIFO CONTENTS    !
                 +--------------------------------+
PCBB+4:          !   FIRST DATA WORD OF BUFFER    !
                 +--------------------------------+
```

TEST 42: HALF-DUPLEX TEST

THE LINK INCLUDES A 'HALF DUPLEX' MODE OF OPERATION. THIS MODE CAN BE ENABLED OR DISABLED THROUGH THE LINK MODE REGISTER. THE OPERATIONAL MICROCODE NORMALLY USES HALF-DUPLEX MODE.

IN THE HALF-DUPLEX MODE, THE LINK WILL NOT RECEIVE MESSAGES ADDRESSED TO ITSELF. INCOMING MESSAGES LOOPED BACK WILL BE IGNORED BY THE RECEIVE STATE MACHINE. THE STATE MACHINE WILL NOT ISSUE A 'RECEIVER DONE' INTERRUPT AND THE BUFFER CAN BE RECOVERED FOR RECEIVING A LATER DATAGRAM.

THIS TEST USES MICROMODULE 'F' MICROTEST #5.
THIS TEST WILL VERIFY THE OPERATION OF HALF-DUPLEX MODE. A DATAGRAM WILL BE TRANSMITTED IN LOOPBACK MODE WITH THE HALF-DUPLEX BIT SET. THE MICROCODE WILL VERIFY THAT THE RECEIVER INTERRUPT DOES NOT OCCUR. THE MICROCODE WILL THEN CLEAR THE HALF-DUPLEX BIT AND LOOP A DATAGRAM AND VERIFY THAT THE ORIGINAL BUFFER WAS RECOVERED.

THIS TEST WILL USE THE PCBB TO PASS INFORMATION. PCBB+0 WILL BE USED TO PASS THE CONTENTS OF THE BUFFER DONE FIFO AFTER THE SECOND DATAGRAM IS RECEIVED. PCBB+4 WILL BE USED TO PASS THE FIRST WORD OF DATA FROM THE RECEIVER BUFFER AFTER THE SECOND DATAGRAM IS TRANSMITTED.

```
                 +--------------------------------+
PCBB+0:          ! CONTENTS OF BUFFER DONE FIFO   !
                 +--------------------------------+
PCBB+2:          ! FIRST DATA WORD OF BUFFER DONE!
```

```
             +-------------------------------+

THE CONTENTS OF THE BUFFER DONE FIFO SHOULD BE 0 AND THE FIRST DATA WORD
SHOULD BE AN ALTERNATING 1'S AND 0'S PATTERN.


        TEST 43: COLLISION TEST

THE RECEIVE STATE MACHINE REACTS TO COLLISIONS ON THE WIRE BY ACTIVATING
THE RETRY LOGIC. THE RETRY LOGIC WAITS AN INTERVAL OF TIME BEFORE ATTEMPTING
TO RETRANSMIT THE DATAGRAM. THE INTERVALS ARE NOT UNIFORM BUT ARE OF
GENERALLY INCREASING PSEUDO-RANDOM DURATION. THE RETRY LOGIC WILL ATTEMPT
TO RETRANSMIT UP TO 15 ADDITIONAL TIMES BEFORE GIVING UP.

THIS TEST WILL VERIFY THAT THE RECEIVE STATE MACHINE RESPONDS TO A COLLISION
AND THAT THE RETRY SEQUENCE IS REPORTED CORRECTLY IN THE TRANSMIT STATUS WORD.

THIS TEST WILL USE MICROMODULE 'G' MICROTEST #1.
THE LINK BOARD CONTATINS DIAGNOSTIC LOGIC THAT ALLOWS COLLSIONS TO BE SIMULATED.
WITH THE FORCE COLLISIONS LOGIC ACTIVATED, THE RETRY HARDWARE CAN BE STEPPED
THROUGH THE RETRY SEQUENCE. THAT IS, EVERY DATAGRAM LOOPED BACK WILL STEP
THE RETRY LOGIC THROUGH ONE STEP OF THE RETRY SEQUENCE. THE RETRY SEQUENCE
CAN BE VERIFIED BY CHECKING THE TRANSMIT BUFFER STATUS WORDS AFTER EACH RETRY
STEP.

THE PCBB WILL BE USED TO PASS PARAMETERS BETWEEN THE MICROCODE AND THE HOST
PROCESSOR. PCBB+0 WILL BE USED TO PASS THE DATA TO BE LOADED INTO THE LINK
MODE WORD. PCBB+2 WILL BE PASSED BACK BY THE MICROCODE, IT IS THE FIRST WORD
OF THE TRANSMIT BUFFER (TRANSMIT STATUS WORD 0). PCBB+4 WILL ALSO BE PASSED
BACK, IT IS TRANSMIT STATUS WORD 1.

THE TRANSMIT STATUS WORDS SHOULD SHOW THE FOLLOWING STATUS:


                STATUS BITS

                WORD 0                 WORD 1
                ---------------        ------
                !             !        !    !
                !             !        !    !
LOOPBACK        ERRS   MORE   ONE      RETRY
  STEP #        (14)   (12)   (11)      (10)
--------        ----   ----   ----     -----

 1               0      0      1         0

 2-15            0      1      0         0

 16              1      0      0         1

THE PCBB IS FORMATTED AS FOLOWS:

                +-------------------+
PCBB+0:         !  LINK MODE WORD   !
                +-------------------+
```

```
PCBB+2:         ! TRANSMIT STATUS WORD 0!
                +-----------------------+
PCBB+4:         ! TRANSMIT STATUS WORD 1!
                +-----------------------+
```

TEST 44: TDR COUNTER TEST

THE DEUNA HAS A COUNTER DESIGNED TO HELP LOCATE FAULTS IN THE COAXIAL CABLE. THE COUNTER IS INITIALIZED WHEN A MESSAGE IS TRANSMITTED AND INCREMENTS AS THE DATAGRAM IS TRANSMITTED. COUNTING WILL STOP IF A COLLISION OCCURS OR THE CARRIER IS LOST. COUNTING ALSO STOPS IF THE 10 BIT COUNTER REACHES ITS MODULUS.

THIS TEST WILL DETERMINE TAHT THE TDR COUNTER VALUE WILL CHANGE AND THAT THE COUNTER IS NOT STUCK.

BECAUSE THE COUNTER COUNTS DURING TRANSMISSION OF A DATAGRAM AND WILL CONTINUE TO COUNT DURING THE TIME THAT THE TRANSMIT STATE MACHINE OPERATES, THE COUNT ACCUMULATED IN THE COUNTER DURING TRANSMISSION IS PROPORTIONAL TO THE LENGTH OF THE DATAGRAM. THIS TEST WILL USE THIS RELATION TO VERIFY THAT THE COUNTER IS NOT STUCK.

THIS TEST USES MICROMODULE 'G' MICROTEST #2.
THE TEST WILL SEND DATAGRAMS OVER THE LOOPBACK. THE LENGTH OF THE DATAGRAM WILL BE VARIED BY USING A INCREASING BYTE COUNT IN THE TRANSMIT BUFFER. AFTER EACH DATAGRAM HAS BEEN LOOPED BACK, THE TRANSMIT BUFFER WORD 1 WILL BE PASSED BACK TO THE HOST TO VERIFY THAT IT IS CORRECT. THE CRITERIA FOR CORRECTNESS WILL BE: INCREASING BYTE COUNTS SHOULD RESULT IN INCREASING TDR VALUES IN TRANSMIT STATUS WORD 1.

THE PCBB WILL BE FORMATED AS FOLLOWS:

```
                +-----------------------+
PCBB+0:         !       BYTE COUNT      !
                +-----------------------+
PCBB+2:         ! TRANSMIT STATUS WORD 1!
                +-----------------------+
```

TEST 45: RETRY LOGIC TEST

THE RETRY LOGIC IS ACTIVATED WHENEVER A COLLISION IS ENCOUNTERED DURING A TRANSMISSION ATTEMPT. THE LINK STOPS TRANSMISSION AND WAITS FOR A PERIOD OF TIME BEFORE ATTEMPTING TO RETRANSMIT.

THE WAIT TIME IS AN INTEGRAL NUMBER OF 'SLOT TIMES'. THE NUMBER COMES FROM A RANDOM NUMBER GENERATOR. THE NUMBER OF SLOT TIMES IS NOT EXACTLY RANDOM SINCE THE RETRY LOGIC WAITS A GENERALLY INCREASING NUMBER OF SLOT TIMES BEFORE TRYING TO RETRANSMIT. THIS TEST WILL VERIFY THAT THE RETRY LOGIC IS CAPABLE OF GENERATING VARIABILITY IN THE DURATION OF THE RETRY WIAT TIMES.

THIS TEST WILL USE MICROMODULE 'G' MICROTEST #3
THE LINK MODULE HAS A DIAGNOSTIC MAINTENANCE FACILITY MAKING IT POSSIBLE TO SINGLE STEP THE RETRY LOGIC THROUGH THE MAXIMUM SIXTEEN RETRY STEPS. THIS FEATURE WILL ALSO MAKE IT POSSIBLE TO MEASURE THE RETRY WAIT INTERVAL.

THE MICROCODE WILL SET THE COLLISION BIT IN THE LINK MODE REGISTER AND AND TRANSMIT A DATAGRAM IN LOOPBACK MODE. THE T-11 WILL COUNT WHILE WAITING FOR THE TRANSMIT STATE MACHINE TO INTERRUPT. THE ACCUMULATED COUNT SHOULD PROVIDE A MEASURE OF TIME TAKEN FOR THE TRANSMISSION ATTEMPT TO OCCUR. SINCE THE COLLISION BIT IS SET, THIS INTERVAL WILL INCLUDE THE RETRY WAIT INTERVAL. THE ACCUMULATED COUNT WILL BE WRITTEN BY THE MICROCODE TO THE PCBB.

THE MICROTEST WILL BE EXECUTED 16 TIMES. AFTER EACH EXECUTION, THE COUNT WILL BE READ FROM THE PCBB AND STORED IN A TABLE. THE TABLE WILL BE SCANNED TO VERIFY THAT THEY ARE NOT ALL THE SAME.

THE PCBB IS FORMATTED AS FOLLOWS:

```
                +------------------------+
PCBB+0:         !       BYTE COUNT       !
                +------------------------+
PCBB+2:         !  TRANSMIT WAIT COUNT   !
                +------------------------+
```

TEST 46: PRINT DEVICE PARAMETERS TEST

THIS TEST PRINTS THE DEFAULT PHYSICAL ADDRESS, THE MICROCODE REVISION AND THE SWITCH PACK SETTINGS.

```
.TITLE  PROGRAM REVISION HISTORY

; DATE      AUTHOR          DESCRIPTION OF CHANGE
;
;3-FEB-83 (MAC001)          ADD THIS SECTION.
;                           REMOVE REDUNDANT .MCALL STATEMENTS TO SVC.
;                           CHANGE INIT CODE TO DELAY A PERIOD OF TIME AFTER A
;                            POWER FAILURE OCCURS TO ALLOW SELF TEST TO FINISH.
;                           INCREASE AMOUNT OF TIME TO WAIT FOR DNI AFTER ISSUING
;                            A RESET TO PCSR0.
;                           ADD PRINTING OF PCSR'S IF ERROR OCCURS.
;                           UPDATE SELF TEST ERROR CODES AND DESCRIPTIONS.
;                           CHANGE TEST 9 TO DECODE PCSR1 IF DNI NEVER HAPPENS.
;                           CHANGE TEST 5 TO NOT CHECK PORT COMMAND FIELD OF PCSR0.
;                           UPDATE 'HEADER' STATEMENT TO REV A-2.
;
;24-MAR-83 (RSJ001)         CHANGE ALL WORD ACCESS TO THE UPPER BYTE OF PCSR0 TO
;                           BYTE ACCESS.  INTRODUCED NEW VARIABLES TO DESCRIBE THE
;                           UPPER BYTE, SAME NAMES WITH B ADDED I.E. DNI -> DNIB.
;                           ADDED ADDRESS STORAGE VARIABLE PCSR0UB AS THE ADDRESS OF
;                           THE UPPER BYTE OF PCSR0.
;                           CHANGED HEADER TO REV B-0.
;
```

```
                                .TITLE PROGRAM HEADER AND TABLES
                                .SBTTL  PROGRAM HEADER

                                        .ENABL  AMA

000000'                                 BGNMOD

                                ;++
                                ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
                                ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
                                ;--

000000'                                 POINTER BGNRPT,BGNAU,BGNDU

000000'                                 HEADER  CZUAA,B,0,0,0,340       ;MAC001
000000'                         L$NAME::
000000'     103                                 .ASCII /C/
000001'     132                                 .ASCII /Z/
000002'     125                                 .ASCII /U/
000003'     101                                 .ASCII /A/
000004'     101                                 .ASCII /A/
000005'     000                                 .BYTE   0
000006'     000                                 .BYTE   0
000007'     000                                 .BYTE   0
000010'                         L$REV::
000010'     102                                 .ASCII  /B/
000011'                         L$DEPO::
000011'     060                                 .ASCII  /0/
000012'                         L$UNIT::
000012' 000000                                  .WORD   0
000014'                         L$TIML::
000014' 000000                                  .WORD   0
000016'                         L$HPCP::
000016' 053256'                                 .WORD   L$HARD
000020'                         L$SPCP::
000020' 000000                                  .WORD   0
000022'                         L$HPTP::
000022' 000262'                                 .WORD   L$HW
000024'                         L$SPTP::
000024' 000000                                  .WORD   0
000026'                         L$LADP::
000026' 000000G                                 .WORD   L$LAST
000030'                         L$STA::
000030' 000000                                  .WORD   0
000032'                         L$CO::
000032' 000000                                  .WORD   0
000034'                         L$DTYP::
000034' 000000                                  .WORD   0
000036'                         L$APT::
000036' 000000                                  .WORD   0
000040'                         L$DTP::
000040' 000124'                                 .WORD   L$DISPATCH
000042'                         L$PRIO::
000042' 000340                                  .WORD   340
000044'                         L$ENVI::
```

```
000044' 000000                     .WORD   0
000046'                    L$EXP1::
000046' 000000                     .WORD   0
000050'                    L$MREV::
000050'    003                     .BYTE   C$REVISION
000051'    003                     .BYTE   C$EDIT
000052'                    L$EF::
000052' 000000                     .WORD   0
000054' 000000                     .WORD   0
000056'                    L$SPC::
000056' 000000                     .WORD   0
000060'                    L$DEVP::
000060' 000700'                    .WORD   L$DVTYP
000062'                    L$REPP::
000062' 021176'                    .WORD   L$RPT
000064'                    L$EXP4::
000064' 000000                     .WORD   0
000066'                    L$EXP5::
000066' 000000                     .WORD   0
000070'                    L$AUT::
000070' 022026'                    .WORD   L$AU
000072'                    L$DUT::
000072' 022020'                    .WORD   L$DU
000074'                    L$LUN::
000074' 000000                     .WORD   0
000076'                    L$DESP::
000076' 000706'                    .WORD   L$DESC
000100'                    L$LOAD::
000100' 104035                     EMT     E$LOAD
000102'                    L$ETP::
000102' 000000                     .WORD   0
000104'                    L$ICP::
000104' 021212'                    .WORD   L$INIT
000106'                    L$CCP::
000106' 021654'                    .WORD   L$CLEAN
000110'                    L$ACP::
000110' 021652'                    .WORD   L$AUTO
000112'                    L$PRT::
000112' 021204'                    .WORD   L$PROT
000114'                    L$TEST::
000114' 000000                     .WORD   0
000116'                    L$DLY::
000116' 000000                     .WORD   0
000120'                    L$HIME::
000120' 000000                     .WORD   0
                                           ;RSJ001
```

```
                                        .SBTTL  DISPATCH TABLE

                                        ;++
                                        ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
                                        ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
                                        ;--

000122'                                         DISPATCH 46
000122' 000056                                                                  .WORD   46
000124'                                                                 L$DISPATCH::
000124' 022076'                                                                 .WORD   T1
000126' 022310'                                                                 .WORD   T2
000130' 022642'                                                                 .WORD   T3
000132' 022754'                                                                 .WORD   T4
000134' 023072'                                                                 .WORD   T5
000136' 023560'                                                                 .WORD   T6
000140' 023642'                                                                 .WORD   T7
000142' 023740'                                                                 .WORD   T8
000144' 024162'                                                                 .WORD   T9
000146' 027124'                                                                 .WORD   T10
000150' 027540'                                                                 .WORD   T11
000152' 030326'                                                                 .WORD   T12
000154' 031006'                                                                 .WORD   T13
000156' 031424'                                                                 .WORD   T14
000160' 031714'                                                                 .WORD   T15
000162' 032422'                                                                 .WORD   T16
000164' 032740'                                                                 .WORD   T17
000166' 033406'                                                                 .WORD   T18
000170' 033764'                                                                 .WORD   T19
000172' 034344'                                                                 .WORD   T20
000174' 034672'                                                                 .WORD   T21
000176' 035202'                                                                 .WORD   T22
000200' 035512'                                                                 .WORD   T23
000202' 036072'                                                                 .WORD   T24
000204' 036452'                                                                 .WORD   T25
000206' 037046'                                                                 .WORD   T26
000210' 037610'                                                                 .WORD   T27
000212' 040206'                                                                 .WORD   T28
000214' 040610'                                                                 .WORD   T29
000216' 041130'                                                                 .WORD   T30
000220' 041520'                                                                 .WORD   T31
000222' 042104'                                                                 .WORD   T32
000224' 042470'                                                                 .WORD   T33
000226' 043130'                                                                 .WORD   T34
000230' 043460'                                                                 .WORD   T35
000232' 044000'                                                                 .WORD   T36
000234' 044350'                                                                 .WORD   T37
000236' 044670'                                                                 .WORD   T38
000240' 045230'                                                                 .WORD   T39
000242' 045620'                                                                 .WORD   T40
000244' 046202'                                                                 .WORD   T41
000246' 046634'                                                                 .WORD   T42
000250' 047240'                                                                 .WORD   T43
000252' 050330'                                                                 .WORD   T44
000254' 051052'                                                                 .WORD   T45
000256' 051466'                                                                 .WORD   T46
```

```
                                  .SBTTL  DEFAULT HARDWARE P-TABLE

                                  ;++
                                  ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
                                  ; THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
                                  ; IS IDENTICAL TO THE STRUCTURE OF THE HARDWARE P-TABLES,
                                  ; AND IS USED AS A "TEMPLATE" FOR BUILDING THE P-TABLES.
                                  ;--

000260'                                 BGNHW   DFPTBL
000260' 000002                                                                  .WORD   L10000-LSHW/2
000262'                                                                  LSHW::
000262'                                                                  DFPTBL::

000262' 000000                          .WORD   0       ;CSR
000264' 000000                          .WORD   0       ;VECTOR

000266'                                 ENDHW
000266'                                                                  L10000:
```

```
                                        .TITLE GLOBAL AREAS
                                        .SBTTL  GLOBAL EQUATES SECTION



                                        ;++
                                        ; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
                                        ; ARE USED IN MORE THAN ONE TEST.
                                        ;--

        000266'                                 EQUALS
                                        ;
                                        ; BIT DIFINITIONS
                                        ;
                100000                  BIT15== 100000
                040000                  BIT14== 40000
                020000                  BIT13== 20000
                010000                  BIT12== 10000
                004000                  BIT11== 4000
                002000                  BIT10== 2000
                001000                  BIT09== 1000
                000400                  BIT08== 400
                000200                  BIT07== 200
                000100                  BIT06== 100
                000040                  BIT05== 40
                000020                  BIT04== 20
                000010                  BIT03== 10
                000004                  BIT02== 4
                000002                  BIT01== 2
                000001                  BIT00== 1
                                        ;
                001000                  BIT9==  BIT09
                000400                  BIT8==  BIT08
                000200                  BIT7==  BIT07
                000100                  BIT6==  BIT06
                000040                  BIT5==  BIT05
                000020                  BIT4==  BIT04
                000010                  BIT3==  BIT03
                000004                  BIT2==  BIT02
                000002                  BIT1==  BIT01
                000001                  BIT0==  BIT00
                                        ;
                                        ; EVENT FLAG DEFINITIONS
                                        ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                        ;
                000040                  EF.START==      32.             ; START COMMAND WAS ISSUED
                000037                  EF.RESTART==    31.             ; RESTART COMMAND WAS ISSUED
                000036                  EF.CONTINUE==   30.             ; CONTINUE COMMAND WAS ISSUED
                000035                  EF.NEW==        29.             ; A NEW PASS HAS BEEN STARTED
                000034                  EF.PWR==        28.             ; A POWER-FAIL/POWER-UP OCCURRED
                                        ;
                                        ;
                                        ; PRIORITY LEVEL DEFINITIONS
                                        ;
                000340                  PRI07== 340
                000300                  PRI06== 300
```

```
        000240                  PRI05== 240
        000200                  PRI04== 200
        000140                  PRI03== 140
        000100                  PRI02== 100
        000040                  PRI01== 40
        000000                  PRI00== 0
                                ;
                                ;OPERATOR FLAG BITS
                                ;
        000004                  EVL==        4
        000010                  LOT==       10
        000020                  ADR==       20
        000040                  IDU==       40
        000100                  ISR==      100
        000200                  UAM==      200
        000400                  BOE==      400
        001000                  PNT==     1000
        002000                  PRI==     2000
        004000                  IXE==     4000
        010000                  IBE==    10000
        020000                  IER==    20000
        040000                  LOE==    40000
        100000                  HOE==   100000

        000077                  SECOND==63.             ;63 LINE CLOCK TICKS = APROX. 1 SECOND
        000001                  SET==   1
        000000                  CLEAR== 0
        004000                  SIZ1K== 4000            ;1K WORDS
        010000                  SIZ2K== SIZ1K*2         ;2K WORDS
        020000                  SIZ4K== SIZ2K*2         ;4K WORDS
        040000                  SIZ8K== SIZ4K*2         ;8K WORDS
        020000                  WCSSIZ==SIZ4K           ;SIZE OF THE DEUNA WRITEABLE CONTROL STORE
        020000                  IOSIZ==SIZ4K            ;SIZE OF THE DEUNA I/O PAGE
        040000                  ROMSIZ==SIZ8K           ;SIZE OF THE DEUNA ROM IN WORDS
        077774                  LINSIZ==SIZ8K*2-4       ;SIZE OF THE DEUNA LINK MEMORY
        000000                  WCSADR==0               ;INTERNAL BASE ADDRESS OF WCS
        020000                  IOADR==WCSADR+WCSSIZ    ;INTERNAL BASE ADDRESS OF THE I/O PAGE FOR THE T11
        040000                  ROMADR==IOADR+IOSIZ     ;INTERNAL BASE ADDRESS OF THE DEUNA ROM
        100000                  LINADR==ROMADR+ROMSIZ   ;INTERNAL BASE ADDRESS OF THE DEUNA LINK MEMORY
        000100                  IE==    BIT6            ;INTERRUPT ENABLE
        177777                  INITH== -1              ;INITIAL HIGH WORD FOR 32 BIT CRC CALCULATOR
        177777                  INITL== -1              ;INITIAL LOW WORD FOR 32 BIT CRC CALCULATOR
        166670                  POLYH== 166670          ;FUNCTION POLYNOMIAL HIGH WORD FOR 32 BIT CRC
        101440                  POLYL== 101440          ;FUNCTION POLYNOMIAL LOW WORD FOR 32 BIT CRC
        120001                  POLY16== 120001         ;FUNCTION POLYNOMIAL FOR 16 BIT CRC
        000000                  DATERR==0               ;DATA ERROR INDICATER FOR LINK MEMORY TESTS
        000001                  ADRERR==1               ;ADDRESS ERROR INDICATOR FOR LINK MEMORY TESTS
        002756                  MAXBYT==1518.           ;MAXIMUM NUMBER OF BYTES RECEIVER CAN HANDLE
        000100                  MINBYT==64.             ;MINIMUM NUMBER OF BYTES RECEIVER CAN HANDLE
        000004                  CRCSIZ==4.              ;SIZE OF CRC
        000001                  INMON=1         ;IN MICROMONITOR STATE *** REMOVE COMMENT MARKS AND
        000002                  INTST=2         ;IN A TEST STATE       *** USE THESE THREE VARIABLES
        000003                  INERR=3         ;IN ERROR STATE        *** FOR ASSEMBLY WITH MACY11
```

```
                        ; PCSR0 - PORT CONTROL AND STATUS  REGISTER 0 (OPERATIONAL MICROCODE DEFINITIONS)

        100000          SERI    ==      BIT15           ; STATUS ERROR INTERRUPT
        000200          SERIB   ==      BIT07           ; STATUS ERROR INTERRUPT BYTE REFERENCE ;RSJ001
        040000          PCEI    ==      BIT14           ; PORT COMMAND ERROR INTERRUPT
        000100          PCEIB   ==      BIT06           ; PORT COMMAND ERROR INTERRUPT BYTE REFERENCE ;RSJ001
        020000          RXI     ==      BIT13           ; RECEIVE RING INTERRUPT
        000040          RXIB    ==      BIT05           ; RECEIVE RING INTERRUPT BYTE REFERENCE ;RSJ001
        010000          TXI     ==      BIT12           ; TRANSMIT RING INTERRUPT
        000020          TXIB    ==      BIT04           ; TRANSMIT RING INTERRUPT BYTE REFERENCE ;RSJ001
        004000          DNI     ==      BIT11           ; DONE INTERRUPT
        000010          DNIB    ==      BIT03           ; DONE INTERRUPT BYTE REFERENCE         ;RSJ001
        002000          RCEI    ==      BIT10           ; RECEIVE BUFFER UNAVAILABLE
        000004          RCEIB   ==      BIT02           ; RECEIVE BUFFER UNAVAILABLE BYTE REFERENCE ;RSJ001
                        ;
        000400          FATI    ==      BIT08           ; FATAL ERROR INTERRUPT
        000001          FATIB   ==      BIT00           ; FATAL ERROR INTERRUPT BYTE REFERENCE  ;RSJ001
        000200          INTR    ==      BIT07           ; INTERRUPT SUMMARY <15:08>
        000100          INTE    ==      BIT06           ; INTERRUPT ENABLE
        000040          RSET    ==      BIT05           ; UNA RESET

                        ;
                        ;PCSR0 - PORT CONTROL AND STATUS REGISTER 0 (DIAGNOSTIC MICROCODE DEFINITIONS)
                        ;
        100000          NPRERR  ==      BIT15           ;T11 NPR TIMEOUT INTERRUPT OCCURRED
        040000          NXMERR  ==      BIT14           ;T11 NON-EXISTANT MEMORY TIMEOUT OCCURRED
        020000          UNIERR  ==      BIT13           ;T11 UNEXPECTED INTERRUPT OCCURRED
        010000          PARERR  ==      BIT12           ;T11 LINK MEMORY PARITY ERROR OCCURRED
                        ;
                        ;PORT COMMANDS <03:00>
        000001          GETPCB  ==      BIT00
        000002          GETCMD  ==      BIT01
        000006          PNOP    ==      BIT01!BIT02
        000003          SLFT    ==      BIT00!BIT01
                        ;
                        ;PCSR1 - PORT CONTROL AND STATUS REGISTER 1
        100000          XPWR    ==      BIT15           ; TRANSCEIVER POWER BAD
        040000          ICAB    ==      BIT14           ; PORT/LINK CABLING OK
                        ;
                        ;SELF TEST ERROR CODE <13:08>
                        ;
        000200          PCTO    ==      BIT07           ; PORT COMMAND TIMEOUT
                        ;
        000010          RMTC    ==      BIT03           ; REMOTE CONSOLE RESERVED
        000400          SFTB0   ==      BIT8            ;FIRST BIT OF SELF TEST FIELD
        001000          SFTB1   ==      BIT9            ;SECOND
        002000          SFTB2   ==      BIT10           ;THIRD
        004000          SFTB3   ==      BIT11           ;FOURTH
        010000          SFTB4   ==      BIT12           ;FIFTH
        020000          SFTB5   ==      BIT13           ;SIXTH
        037400          SFT     ==      BIT8!BIT9!BIT10!BIT11!BIT12!BIT13       ;SELF TEST FIELD
                        ;
                        ;PORT STATE <02:00>
                        ;
        000007          PSTATE  ==      BIT00!BIT01!BIT02 ;PORT STATE BITS OF PCSR1
```

```
000000        RESET   ==      0                 ;RESET STATE (NOT REALLY A STATE)
000001        PRILD   ==      BIT00             ; PRIMARY LOAD STATE
000002        READY   ==      BIT01             ;READY STATE
000003        RUN     ==      BIT00!BIT01       ;RUNNING STATE
000005        UNIHLT  ==      BIT00!BIT02       ;UNIBUS HALTED STATE
000006        NIHLT   ==      BIT01!BIT02       ;NI HALTED STATE
000007        NIUNI   ==      BIT00!BIT01!BIT02 ;UNIBUS AND NI HALTED STATE
                      ;
                      ;DESCRIPTOR RING DEFINITIONS
100000        OWN     ==      BIT15
040000        ERRS    ==      BIT14
001000        STP     ==      BIT09
000400        ENP     ==      BIT08
                      ;
                      ;PORT FUNCTION CODES
                      ;

000000        PFNOP1  ==      0           ;NOP PORT FUNCTION #1
000001        LASM    ==      1           ;LOAD AND START MICROADDRESS
000002        RDPA    ==      2           ;READ DEFAULT PHYSICAL ADDRESS
000003        PFNOP2  ==      3           ;NOP PORT FUNCTION #2
000004        RPA     ==      4           ;READ PHYSICAL ADDRESS
000005        WPA     ==      5           ;WRITE PHYSICAL ADDRESS
000006        RMAL    ==      6           ;READ MULTICAST ADDRESS LIST
000007        WMAL    ==      7           ;WRITE MULTICAST ADDRESS LIST
000010        RRF     ==      10          ;READ RING FORMAT
000011        WRF     ==      11          ;WRITE RING FORMAT
000012        RC      ==      12          ;READ COUNTERS
000013        RACC    ==      13          ;READ AND CLEAR COUNTERS
000014        RTM     ==      14          ;READ THE MODE
000015        WTM     ==      15          ;WRITE THE MODE
000016        RPS     ==      16          ;READ PORT STATUS
000017        RACPS   ==      17          ;READ AND CLEAR PORT STATUS
000020        DIM     ==      20          ;DUMP INTERNAL MEMORY
000021        LIM     ==      21          ;LOAD INTERNAL MEMORY
000022        RSIDP   ==      22          ;READ SYSTEM ID PARAMETERS
000023        WSIDP   ==      23          ;WRITE SYSTEM ID PARAMETERS
000024        RLSA    ==      24          ;READ LOAD SERVER ADDRESS
000025        WLSA    ==      25          ;WRITE LOAD SERVER ADDRESS
```

```
                                .SBTTL  GLOBAL DATA SECTION
                                ;++
                                ; THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
                                ; IN MORE THAN ONE TEST.
                                ;--
000266' 000000                  UNACSR::.WORD   0       ;CSR
000270' 000000                  UNAVEC::.WORD   0       ;VECTOR
000272' 000000                  UNAPRI::.WORD   0       ;PRIORITY
000274'                         CLKTAB::
000274' 000000                  CLKCSR::.WORD   0       ;LINE CLOCK STATUS REGISTER
000276' 000000                  CLKBR:: .WORD   0       ;LINE CLOCK PRIORITY
000300' 000000                  CLKVEC::.WORD   0       ;LINE CLOCK VECTOR
000302' 000000                  CLKFRE::.WORD   0       ;LINE CLOCK FREQUENCY
000304' 000000                  CSRNUM::.WORD   0       ;PCSR NUMBER
000306' 000000                  BITNUM::.WORD   0       ;BIT NUMBER
000310' 000000                  BITNAM::.WORD   0       ;POINTER TO BIT NAME ASCII STRING
000312' 000000                  BITSTA::.WORD   0       ;POINTER TO BIT STATUS ASCII STRING
000314' 000000                  PWHEN:: .WORD   0       ;POINTER TO 'BEFORE' OR 'AFTER' ASCII STRING
000316' 000000                  PCOMND::.WORD   0       ;POINTER TO PORT COMMAND ASCII STRING
000320' 000000                  MICMOD::.WORD   0       ;POINTER TO MICROCODE MODULE # LOADED
000322' 000000                  FRESIZ::.WORD   0       ;POINTER TO WORD CONTAINING SIZE OF FREE MEMORY
000324' 000000                  FREMEM::.WORD   0       ;POINTER TO FREE MEMORY SPACE
000326' 000000                  MICRO:: .WORD   0       ;CURRENT MICROCODE MODULE LOADED
000330' 000000                  UNIT::  .WORD   0       ;CURRENT UNIT NUMBER BEING TESTED
000332' 000000                  METER:: .WORD   0       ;CLOCK TICKS
000334' 021040                  SWADDR::.WORD   21040   ;INTERNAL ADDRESS OF SWITCH PACK
000336' 000000                  PCSR0:: .WORD   0       ;PORT CONTROL AND STATUS REGISTER 0
000340' 000000                  PCSR1:: .WORD   0       ;PORT CONTROL AND STATUS REGISTER 1
000342' 000000                  PCSR2:: .WORD   0       ;PORT CONTROL AND STATUS REGISTER 2
000344' 000000                  PCSR3:: .WORD   0       ;PORT CONTROL AND STATUS REGISTER 3
000346' 000000                  PCSR0UB:: .WORD 0       ;PORT CONTROL AND STATUS REGISTER 0 UPPER BYTE ;RSJ001
000350' 000000                  PCSR0C::.WORD   0       ;PCSR0 CONTENTS
000352' 000000                  PCSR1C::.WORD   0       ;PCSR1 CONTENTS
000354' 000000                  PCSR2C::.WORD   0       ;PCSR2 CONTENTS
000356' 000000                  PCSR3C::.WORD   0       ;PCSR3 CONTENTS
000360'                         BNAMT0::                ;TABLE OF POINTERS TO BIT NAME MNEMONICS FOR PCSR0
000360' 001271'                         $BIT0
000362' 001263'                         $BIT1
000364' 001255'                         $BIT2
000366' 001247'                         $BIT3
000370' 001241'                         $BIT4
000372' 001054'                         $RSET
000374' 001043'                         $INTE
000376' 001032'                         $INTR
000400' 001021'                         $FATI
```

```
000402' 001203'                 $BIT9
000404' 001010'                 $RCEI
000406' 001000'                 $DNI
000410' 000770'                 $TXI
000412' 000760'                 $RXI
000414' 000747'                 $PCEI
000416' 000736'                 $SERI
000420'                 BNAMT1::                        ;TABLE OF POINTERS TO BIT NAME MNEMONICS FOR PCSR1
000420' 000000                  .WORD   0
000422' 000000                  .WORD   0
000424' 000000                  .WORD   0
000426' 001120'                 $RMTC
000430' 001241'                 $BIT4
000432' 001233'                 $BIT5
000434' 001225'                 $BIT6
000436' 001107'                 $PCTO
000440' 000000                  .WORD   0
000442' 000000                  .WORD   0
000444' 000000                  .WORD   0
000446' 000000                  .WORD   0
000450' 000000                  .WORD   0
000452' 000000                  .WORD   0
000454' 001076'                 $ICAB
000456' 001065'                 $XPWR

000460'                 BNAMT2::                        ;TABLE OF GENERIC BIT NAME MNEMONICS
000460' 001271'                 $BIT0
000462' 001263'                 $BIT1
000464' 001255'                 $BIT2
000466' 001247'                 $BIT3
000470' 001241'                 $BIT4
000472' 001233'                 $BIT5
000474' 001225'                 $BIT6
000476' 001217'                 $BIT7
000500' 001211'                 $BIT8
000502' 001203'                 $BIT9
000504' 001174'                 $BIT10
000506' 001165'                 $BIT11
000510' 001156'                 $BIT12
000512' 001147'                 $BIT13
000514' 001140'                 $BIT14
000516' 001131'                 $BIT15

000520'                 PATERN::
000520' 125252          PAT1::  .WORD   ^B1010101010101010
000522' 052525          PAT2::  .WORD   ^B0101010101010101
000524' 031463          PAT3::  .WORD   ^B0011001100110011
000526' 007417          PAT4::  .WORD   ^B0000111100001111
000530' 000377          PAT5::  .WORD   ^B0000000011111111
000532' 177777          PAT6::  .WORD   ^B1111111111111111

000534' 125252          SPAT1:: .WORD   ^B1010101010101010
000536' 125252                  .WORD   ^B1010101010101010
000540' 125252                  .WORD   ^B1010101010101010
000542' 052525          SPAT2:: .WORD   ^B0101010101010101
000544' 052525                  .WORD   ^B0101010101010101
```

```
000546' 052525                  .WORD   ^B0101010101010101
000550' 031463          SPAT3:: .WORD   ^B0011001100110011
000552' 031463                  .WORD   ^B0011001100110011
000554' 031463                  .WORD   ^B0011001100110011
000556' 007417          SPAT4:: .WORD   ^B0000111100001111
000560' 007417                  .WORD   ^B0000111100001111
000562' 007417                  .WORD   ^B0000111100001111
000564' 000377          SPAT5:: .WORD   ^B0000000011111111
000566' 000377                  .WORD   ^B0000000011111111
000570' 000377                  .WORD   ^B0000000011111111
000572' 177777          SPAT6:: .WORD   ^B1111111111111111
000574' 000000                  .WORD   ^B0000000000000000
000576' 177777                  .WORD   ^B1111111111111111
000600' 177777          SPAT7:: .WORD   ^B1111111111111111
000602' 000377                  .WORD   ^B0000000011111111
000604' 000000                  .WORD   ^B0000000000000000

000606' 000000          PCBB::  .WORD   0       ;PORT CONTROL BLOCK
000610' 000000                  .WORD   0       ;+2
000612' 000000                  .WORD   0       ;+4
000614' 000000                  .WORD   0       ;+6

000616' 000000          UDBB::  .WORD   0       ;UNIBUS DATA BLOCK BASE
000620' 000000                  .WORD   0       ;+2
000622' 000000                  .WORD   0       ;+4
000624' 000000                  .WORD   0       ;+6

000626' 000020          CNTTAB::.BLKW   16.     ;HOLDS THE 16 RETRY WAIT COUNTS FOR TEST 39

000666' 000000          NEXMEM::.WORD   0       ;NON-EXISTANT MEMORY FLAG
000670' 000000          ERRINT::.WORD   0       ;HOLDS PCSRO INTERRUPT FLAG BITS FROM MICROCODE
000672' 000000          UNAINT::.WORD   0       ;UNA INTERRUPT FLAG
000674' 000000          FRSTIM::.WORD   0       ;FLAG INDICATING FIRST TIME THROUGH TESTS FOR ALL UNITS
000676' 000000          CPUPRI::.WORD   0       ;CPU PRIORITY BEFORE INTERRUPT OCCURRED

                        .EVEN
```

```
                                        .SBTTL  GLOBAL TEXT SECTION

                                        ;++
                                        ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
                                        ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
                                        ; MORE THAN ONE TEST.
                                        ;--

                                        ;
                                        ; NAMES OF DEVICES SUPPORTED BY PROGRAM
                                        ;
 000700'                                        DEVTYP  <DEUNA>
 000700'                                                                        L$DVTYP::
 000700' 042504  047125  000101                                                         .ASCIZ  /DEUNA/
                                                                                        .EVEN


                                        ;
                                        ; TEST DESCRIPTION
                                        ;
 000706'                                        DESCRIPT        <DEUNA REPAIR DIAGNOSTIC>
 000706'                                                                        L$DESC::
 000706' 042504  047125  020101                                                         .ASCIZ  /DEUNA REPAIR DI
 000714' 042522  040520  051111
 000722' 042040  040511  047107
 000730' 051517  044524  000103
                                                                                        .EVEN

 000736' 042523  044522  041040  $SERI:: .ASCIZ  /SERI BIT/
 000744' 052111     000
 000747'    120  042503  020111  $PCEI:: .ASCIZ  /PCEI BIT/
 000754' 044502  000124
 000760' 054122  020111  044502  $RXI::  .ASCIZ  /RXI BIT/
 000766' 000124
 000770' 054124  020111  044502  $TXI::  .ASCIZ  /TXI BIT/
 000776' 000124
 001000' 047104  020111  044502  $DNI::  .ASCIZ  /DNI BIT/
 001006' 000124
 001010' 041522  044505  041040  $RCEI:: .ASCIZ  /RCEI BIT/
 001016' 052111     000
 001021'    106  052101  020111  $FATI:: .ASCIZ  /FATI BIT/
 001026' 044502  000124
 001032' 047111  051124  041040  $INTR:: .ASCIZ  /INTR BIT/
 001040' 052111     000
 001043'    111  052116  020105  $INTE:: .ASCIZ  /INTE BIT/
 001050' 044502  000124
 001054' 051522  052105  041040  $RSET:: .ASCIZ  /RSET BIT/
 001062' 052111     000
 001065'    130  053520  020122  $XPWR:: .ASCIZ  /XPWR BIT/
 001072' 044502  000124
 001076' 041511  041101  041040  $ICAB:: .ASCIZ  /ICAB BIT/
 001104' 052111     000
 001107'    120  052103  020117  $PCTO:: .ASCIZ  /PCTO BIT/
```

```
001114' 044502  000124
001120' 046522  041524  041040  $RMTC:: .ASCIZ  /RMTC BIT/
001126' 052111     000
001131'    102  052111  030440  $BIT15:::.ASCIZ  /BIT 15/
001136' 000065
001140' 044502  020124  032061  $BIT14:::.ASCIZ  /BIT 14/
001146'    000
001147'    102  052111  030440  $BIT13:::.ASCIZ  /BIT 13/
001154' 000063
001156' 044502  020124  031061  $BIT12:::.ASCIZ  /BIT 12/
001164'    000
001165'    102  052111  030440  $BIT11:::.ASCIZ  /BIT 11/
001172' 000061
001174' 044502  020124  030061  $BIT10:::.ASCIZ  /BIT 10/
001202'    000
001203'    102  052111  034440  $BIT9:: .ASCIZ  /BIT 9/
001210'    000
001211'    102  052111  034040  $BIT8:: .ASCIZ  /BIT 8/
001216'    000
001217'    102  052111  033440  $BIT7:: .ASCIZ  /BIT 7/
001224'    000
001225'    102  052111  033040  $BIT6:: .ASCIZ  /BIT 6/
001232'    000
001233'    102  052111  032440  $BIT5:: .ASCIZ  /BIT 5/
001240'    000
001241'    102  052111  032040  $BIT4:: .ASCIZ  /BIT 4/
001246'    000
001247'    102  052111  031440  $BIT3:: .ASCIZ  /BIT 3/
001254'    000
001255'    102  052111  031040  $BIT2:: .ASCIZ  /BIT 2/
001262'    000
001263'    102  052111  030440  $BIT1:: .ASCIZ  /BIT 1/
001270'    000
001271'    102  052111  030040  $BIT0:: .ASCIZ  /BIT 0/
001276'    000
001277'    116  052117  051440  $NSET:: .ASCIZ  /NOT SET/
001304' 052105     000
001307'    123  052105     000  $SET::  .ASCIZ  /SET/
001313'    116  052117  041440  $NCLR:: .ASCIZ  /NOT CLEAR/
001320' 042514  051101     000
001325'    103  042514  051101  $CLR::  .ASCIZ  /CLEAR/
001332'    000
001333'    102  043105  051117  $BEFOR:::.ASCIZ  /BEFORE/
001340' 000105
001342' 043101  042524  000122  $AFTER:::.ASCIZ  /AFTER/
001350' 042507  050124  041103  $GTPCB:::.ASCIZ  /GETPCB/
001356'    000
001357'    107  052105  041440  $GTCMD:::.ASCIZ  /GET COMMAND/
001364' 046517  040515  042116
001372'    000
001373'    123  046105  020106  $SLFT:: .ASCIZ  /SELF TEST/
001400' 042524  052123     000
001405'    116  050117     000  $NOP::  .ASCIZ  /NOP/
001411'    123  040524  052122  $STRT:: .ASCIZ  /START/
001416'    000
001417'    104  046505  047101  $PDNDM:::.ASCIZ  /DEMAND POLL/
```

```
001424' 020104  047520  046114
001432'    000
001433'    123  047524  000120  $STOP:: .ASCIZ  /STOP/
001440' 042522  042523  000124  $RESET::.ASCIZ  /RESET/
001446' 040504  040524  042440  $DATER::.ASCIZ  /DATA ERROR/
001454' 051122  051117     000
001461'    101  042104  042522  $ADRER::.ASCIZ  /ADDRESSING ERROR/
001466' 051523  047111  020107
001474' 051105  047522  000122
001502' 040520  044522  054524  $PARER::.ASCIZ  /PARITY ERROR/
001510' 042440  051122  051117
001516'    000

                                ;
                                ; FORMAT STATEMENTS USED IN PRINT CALLS
                                ;

001517'    045  022516  052501  UNLOD:: .ASCIZ  /%N%AUNABLE TO LOAD MICROCODE MODULE %T%N/
001524' 040516  046102  020105
001532' 047524  046040  040517
001540' 020104  044515  051103
001546' 041517  042117  020105
001554' 047515  052504  042514
001562' 022440  022524  000116
001570' 042522  044507  052123  RACERR::.ASCIZ  /REGISTER ACCESS TEST FAILED/
001576' 051105  040440  041503
001604' 051505  020123  042524
001612' 052123  043040  044501
001620' 042514  000104
001624' 042522  042523  020124  RSETER::.ASCIZ  /RESET TEST FAILED/
001632' 042524  052123  043040
001640' 044501  042514  000104
001646' 042522  044507  052123  RRWER:: .ASCIZ  'REGISTER READ/WRITE TEST FAILED'
001654' 051105  051040  040505
001662' 027504  051127  052111
001670' 020105  042524  052123
001676' 043040  044501  042514
001704' 000104
001706' 047516  020120  042524  NOPERR::.ASCIZ  /NOP TEST FAILED/
001714' 052123  043040  044501
001722' 042514  000104
001726' 042523  043114  052040  SLFTST::.ASCIZ  /SELF TEST FAILED/
001734' 051505  020124  040506
001742' 046111  042105     000
001747'    122  046517  042040  ROMDMP::.ASCIZ  /ROM DUMP TEST FAILED/
001754' 046525  020120  042524
001762' 052123  043040  044501
001770' 042514  000104
001774' 041527  020123  047514  DATALD::.ASCIZ  'WCS LOAD/DUMP TEST FAILED'
002002' 042101  042057  046525
002010' 020120  042524  052123
002016' 043040  044501  042514
002024' 000104
002026' 047514  042101  040440  LASFT:: .ASCIZ  /LOAD AND START FUNCTION TEST FAILED/
002034' 042116  051440  040524
002042' 052122  043040  047125
```

```
002050' 052103  047511  020116
002056' 042524  052123  043040
002064' 044501  042514  000104
002072' 041527  020123  042515  WCSMEM::.ASCIZ  /WCS MEMORY TEST FAILED/
002100' 047515  054522  052040
002106' 051505  020124  040506
002114' 046111  042105     000
002121'    111  052116  051105  INTVEC::.ASCIZ  /INTERRUPT VECTOR TEST FAILED/
002126' 052522  052120  053040
002134' 041505  047524  020122
002142' 042524  052123  043040
002150' 044501  042514  000104
002156' 041520  051123  020060  INTBIT::.ASCIZ  /PCSRO INTERRUPT BIT TEST FAILED/
002164' 047111  042524  051122
002172' 050125  020124  044502
002200' 020124  042524  052123
002206' 043040  044501  042514
002214' 000104
002216' 044524  042515  020122  TIMTST::.ASCIZ  /TIMER TEST FAILED/
002224' 042524  052123  043040
002232' 044501  042514  000104
002240' 044514  045516  046440  LNKMEM::.ASCIZ  /LINK MEMORY TEST FAILED/
002246' 046505  051117  020131
002254' 042524  052123  043040
002262' 044501  042514  000104
002270' 046504  020101  052047  DMATO:: .ASCIZ  /DMA 'TO' ADDRESS REGISTER TEST FAILED/
002276' 023517  040440  042104
002304' 042522  051523  051040
002312' 043505  051511  042524
002320' 020122  042524  052123
002326' 043040  044501  042514
002334' 000104
002336' 046504  020101  043047  DMAFRM::.ASCIZ  /DMA 'FROM' ADDRESS REGISTER TEST FAILED/
002344' 047522  023515  040440
002352' 042104  042522  051523
002360' 051040  043505  051511
002366' 042524  020122  042524
002374' 052123  043040  044501
002402' 042514  000104
002406' 046504  020101  046102  DMABLK::.ASCIZ  /DMA BLOCK TRANSFER TEST FAILED/
002414' 041517  020113  051124
002422' 047101  043123  051105
002430' 052040  051505  020124
002436' 040506  046111  042105
002444'    000
002445'    124  040522  051516  TRNDON::.ASCIZ  /TRANSMIT DONE TEST FAILED/
002452' 044515  020124  047504
002460' 042516  052040  051505
002466' 020124  040506  046111
002474' 042105     000
002477'    122  041505  044505  RCVDON::.ASCIZ  /RECEIVER DONE TEST FAILED/
002504' 042526  020122  047504
002512' 042516  052040  051505
002520' 020124  040506  046111
002526' 042105     000
002531'    104  052101  020101  DBFRAM::.ASCIZ  /DATA BYTE FRAMING TEST FAILED/
```

```
002536' 054502  042524  043040
002544' 040522  044515  043516
002552' 052040  051505  020124
002560' 040506  046111  042105
002566'    000
002567'    104  052101  020101  DWFRAM::.ASCIZ  /DATA WORD FRAMING TEST FAILED/
002574' 047527  042122  043040
002602' 040522  044515  043516
002610' 052040  051505  020124
002616' 040506  046111  042105
002624'    000
002625'    104  052101  020101  DPPAT:: .ASCIZ  /DATA PATH PATTERN TEST FAILED/
002632' 040520  044124  050040
002640' 052101  042524  047122
002646' 052040  051505  020124
002654' 040506  046111  042105
002662'    000
002663'    123  040524  052524  STAMUX::.ASCIZ  /STATUS MUX VERIFICATION TEST FAILED/
002670' 020123  052515  020130
002676' 042526  044522  044506
002704' 040503  044524  047117
002712' 052040  051505  020124
002720' 040506  046111  042105
002726'    000
002727'    114  047111  020113  LNKBYT::.ASCIZ  /LINK BYTE COUNTER TEST FAILED/
002734' 054502  042524  041440
002742' 052517  052116  051105
002750' 052040  051505  020124
002756' 040506  046111  042105
002764'    000
002765'    114  047111  020113  ODDBYT::.ASCIZ  /LINK ODD BYTE COUNTER TEST FAILED/
002772' 042117  020104  054502
003000' 042524  041440  052517
003006' 052116  051105  052040
003014' 051505  020124  040506
003022' 046111  042105     000
003027'    114  047111  020113  MAXCNT::.ASCIZ  /LINK MAXIMUM BYTE COUNT TEST FAILED/
003034' 040515  044530  052515
003042' 020115  054502  042524
003050' 041440  052517  052116
003056' 052040  051505  020124
003064' 040506  046111  042105
003072'    000
003073'    106  043111  020117  FIFTST::.ASCIZ  /FIFO TEST FAILED/
003100' 042524  052123  043040
003106' 044501  042514  000104
003114' 042522  042503  053111  RLNKAD::.ASCIZ  /RECEIVER LINK MEMORY ADDRESS TEST FAILED/
003122' 051105  046040  047111
003130' 020113  042515  047515
003136' 054522  040440  042104
003144' 042522  051523  052040
003152' 051505  020124  040506
003160' 046111  042105     000
003165'    124  040522  051516  TLNKAD::.ASCIZ  /TRANSMITTER LINK MEMORY ADDRESS TEST FAILED/
003172' 044515  052124  051105
003200' 046040  047111  020113
```

```
003206' 042515  047515  054522
003214' 040440  042104  042522
003222' 051523  052040  051505
003230' 020124  040506  046111
003236' 042105     000
003241'    114  047111  020113  LNKARB::.ASCIZ  /LINK MEMORY ARBITRATION TEST FAILED/
003246' 042515  047515  054522
003254' 040440  041122  052111
003262' 040522  044524  047117
003270' 052040  051505  020124
003276' 040506  046111  042105
003304'    000
003305'    123  040524  044524  STAPAT::.ASCIZ  /STATION ADDRESS PATTERN TEST FAILED/
003312' 047117  040440  042104
003320' 042522  051523  050040
003326' 052101  042524  047122
003334' 052040  051505  020124
003342' 040506  046111  042105
003350'    000
003351'    123  040524  044524  STAREJ::.ASCIZ  /STATION ADDRESS REJECTION TEST FAILED/
003356' 047117  040440  042104
003364' 042522  051523  051040
003372' 045105  041505  044524
003400' 047117  052040  051505
003406' 020124  040506  046111
003414' 042105     000
003417'    123  040524  044524  STAPOS::.ASCIZ  /STATION ADDRESS RAM POSITION TEST FAILED/
003424' 047117  040440  042104
003432' 042522  051523  051040
003440' 046501  050040  051517
003446' 052111  047511  020116
003454' 042524  052123  043040
003462' 044501  042514  000104
003470' 052515  052114  041511  MUCAST::.ASCIZ  /MULTICAST ADDRESS TEST FAILED/
003476' 051501  020124  042101
003504' 051104  051505  020123
003512' 042524  052123  043040
003520' 044501  042514  000104
003526' 051103  020103  040504  CRCDAT::.ASCIZ  /CRC DATA PATTERN TEST FAILED/
003534' 040524  050040  052101
003542' 042524  047122  052040
003550' 051505  020124  040506
003556' 046111  042105     000
003563'    103  041522  042440  CRCERR::.ASCIZ  /CRC ERROR TEST FAILED/
003570' 051122  051117  052040
003576' 051505  020124  040506
003604' 046111  042105     000
003611'    103  041522  050040  CRCPAT::.ASCIZ  /CRC PATTERN LENGTH TEST FAILED/
003616' 052101  042524  047122
003624' 046040  047105  052107
003632' 020110  042524  052123
003640' 043040  044501  042514
003646' 000104
003650' 042522  042503  053111  RBRRUN::.ASCIZ  /RECEIVE BUFFER RECOVERY - RUNT TEST FAILED/
003656' 020105  052502  043106
003664' 051105  051040  041505
```

```
003672' 053117  051105  020131
003700' 020055  052522  052116
003706' 052040  051505  020124
003714' 040506  046111  042105
003722'    000
003723'    110  046101  026506  HAFDUP::.ASCIZ  /HALF-DUPLEX TEST FAILED/
003730' 052504  046120  054105
003736' 052040  051505  020124
003744' 040506  046111  042105
003752'    000
003753'    103  046117  044514  COLTST::.ASCIZ  /COLLISION TEST FAILED/
003760' 044523  047117  052040
003766' 051505  020124  040506
003774' 046111  042105     000
004001'    124  051104  041440  TDRCNT::.ASCIZ  /TDR COUNTER TEST FAILED/
004006' 052517  052116  051105
004014' 052040  051505  020124
004022' 040506  046111  042105
004030'    000
004031'    122  052105  054522  RETLOG::.ASCIZ  /RETRY LOGIC TEST FAILED/
004036' 046040  043517  041511
004044' 052040  051505  020124
004052' 040506  046111  042105
004060'    000
004061'    125  040516  046102  PRTPAR::.ASCIZ  /UNABLE TO PRINT DEVICE PARAMETERS FOR THIS UNIT/
004066' 020105  047524  050040
004074' 044522  052116  042040
004102' 053105  041511  020105
004110' 040520  040522  042515
004116' 042524  051522  043040
004124' 051117  052040  044510
004132' 020123  047125  052111
004140'    000
004141'    045  050101  051503  FORM1:: .ASCIZ  /%APCSR%D1%A DOES NOT EXIST%N/
004146' 022522  030504  040445
004154' 042040  042517  020123
004162' 047516  020124  054105
004170' 051511  022524  000116
004176' 040445  041520  051123  FORM2:: .ASCIZ  /%APCSR%D1%A BIT %Z2%A IS %T%N/
004204' 042045  022461  020101
004212' 044502  020124  055045
004220' 022462  020101  051511
004226' 022440  022524  000116
004234' 040445  041520  051123  FORM3:: .ASCIZ  /%APCSR%D1%A FAILED DATA PATTERN TEST%N/
004242' 042045  022461  020101
004250' 040506  046111  042105
004256' 042040  052101  020101
004264' 040520  052124  051105
004272' 020116  042524  052123
004300' 047045     000
004303'    045  042101  052101  FORM4:: .ASCIZ  /%ADATA WRITTEN: %O6%A DATA READ: %O6%N/
004310' 020101  051127  052111
004316' 042524  035116  022440
004324' 033117  040445  042040
004332' 052101  020101  042522
004340' 042101  020072  047445
```

```
004346' 022466  000116
004352' 040445  040503  047116  FORM5:: .ASCIZ  /%ACANNOT CLEAR INTE BIT IN PCSR0%N/
004360' 052117  041440  042514
004366' 051101  044440  052116
004374' 020105  044502  020124
004402' 047111  050040  051503
004410' 030122  047045     000
004415'    045  022524  030523  FORM6:: .ASCIZ  /%T%S1%T%S1%T%S1%T%A PORT COMMAND WAS ISSUED%N/
004422' 052045  051445  022461
004430' 022524  030523  052045
004436' 040445  050040  051117
004444' 020124  047503  046515
004452' 047101  020104  040527
004460' 020123  051511  052523
004466' 042105  047045     000
004473'    045  041501  041522  FORM7:: .ASCIZ  /%ACRC CHECK ERROR ON DATA DUMPED FROM DEUNA ROM%N/
004500' 041440  042510  045503
004506' 042440  051122  051117
004514' 047440  020116  040504
004522' 040524  042040  046525
004530' 042520  020104  051106
004536' 046517  042040  052505
004544' 040516  051040  046517
004552' 047045     000
004555'    045  042101  044516  FORM8:: .ASCIZ  /%ADNI BIT DID NOT CLEAR AFTER WRITING A 1 TO IT%N/
004562' 041040  052111  042040
004570' 042111  047040  052117
004576' 041440  042514  051101
004604' 040440  052106  051105
004612' 053440  044522  044524
004620' 043516  040440  030440
004626' 052040  020117  052111
004634' 047045     000
004637'    045  050101  051503  FORM9:: .ASCIZ  /%APCSR%D1%S1%T%S1%T%S1%T%A RESET WAS ISSUED%N/
004644' 022522  030504  051445
004652' 022461  022524  030523
004660' 052045  051445  022461
004666' 022524  020101  042522
004674' 042523  020124  040527
004702' 020123  051511  052523
004710' 042105  047045     000
004715'    045  042101  052505  FORM10::.ASCIZ  /%ADEUNA NOT IN READY STATE AFTER RESET WAS ISSUED%N/
004722' 040516  047040  052117
004730' 044440  020116  042522
004736' 042101  020131  052123
004744' 052101  020105  043101
004752' 042524  020122  042522
004760' 042523  020124  040527
004766' 020123  051511  052523
004774' 042105  047045     000
005001'    045  042101  052101  FORM11::.ASCIZ  /%ADATA COMPARE ERROR%N/
005006' 020101  047503  050115
005014' 051101  020105  051105
005022' 047522  022522  000116
005030' 040445  052502  043106  FORM12::.ASCIZ  /%ABUFFER ADDRESS: %O6%S1%ADATA SB: %O6%S1%ADATA WAS: %O6%N/
005036' 051105  040440  042104
```

```
005044' 042522 051523 020072
005052' 047445 022466 030523
005060' 040445 040504 040524
005066' 051440 035102 022440
005074' 033117 051445 022461
005102' 042101 052101 020101
005110' 040527 035123 022440
005116' 033117 047045    000
005123'    045 030501 020066  FORM13::.ASCIZ  /%A16 BIT CRC SHOULD BE: 0%S2%AWAS: %06%N/
005130' 044502 020124 051103
005136' 020103 044123 052517
005144' 042114 041040 035105
005152' 030040 051445 022462
005160' 053501 051501 020072
005166' 047445 022466 000116
005174' 040445 040504 040524  FORM15::.ASCIZ  /%ADATA WRITTEN TO PCSR1 FROM T11 WAS: %06%N/
005202' 053440 044522 052124
005210' 047105 052040 020117
005216' 041520 051123 020061
005224' 051106 046517 052040
005232' 030461 053440 051501
005240' 020072 047445 022466
005246' 000116
005250' 040445 040504 040524  FORM16::.ASCIZ  /%ADATA READ FROM PCSR1 FROM UNIBUS WAS: %06%N/
005256' 051040 040505 020104
005264' 051106 046517 050040
005272' 051503 030522 043040
005300' 047522 020115 047125
005306' 041111 051525 053440
005314' 051501 020072 047445
005322' 022466 000116
005326' 040445 047516 044440  FORM17::.ASCIZ  /%ANO INTERRUPT AFTER NOP PORT COMMAND WAS ISSUED%N/
005334' 052116 051105 052522
005342' 052120 040440 052106
005350' 051105 047040 050117
005356' 050040 051117 020124
005364' 047503 046515 047101
005372' 020104 040527 020123
005400' 051511 052523 042105
005406' 047045    000
005411'    045 052501 040516  FORM18::.ASCIZ  /%AUNA DID NOT INTERRUPT AT CORRECT PRIORITY%N/
005416' 042040 042111 047040
005424' 052117 044440 052116
005432' 051105 052522 052120
005440' 040440 020124 047503
005446' 051122 041505 020124
005454' 051120 047511 044522
005462' 054524 047045    000
005467'    045 047101 020117  FORM19::.ASCIZ  /%ANO INTERRUPT AFTER T11 SET %T%A IN PCSR0%N/
005474' 047111 042524 051122
005502' 050125 020124 043101
005510' 042524 020122 030524
005516' 020061 042523 020124
005524' 052045 040445 044440
005532' 020116 041520 051123
005540' 022460 000116
```

```
005544' 040445 044524 042515 FORM20::.ASCIZ /%ATIMER DID NOT INTERRUPT T11%N/
005552' 020122 044504 020104
005560' 047516 020124 047111
005566' 042524 051122 050125
005574' 020124 030524 022461
005602' 000116
005604' 040445 044524 042515 FORM21::.ASCIZ /%ATIMER DID NOT INTERRUPT WHEN EXPECTED%N/
005612' 020122 044504 020104
005620' 047516 020124 047111
005626' 042524 051122 050125
005634' 020124 044127 047105
005642' 042440 050130 041505
005650' 042524 022504 000116
005656' 040445 054105 042520 FORM22::.ASCIZ /%AEXPECTED INTERRUPT BETWEEN 8 AND 12 SECONDS%N/
005664' 052103 042105 044440
005672' 052116 051105 052522
005700' 052120 041040 052105
005706' 042527 047105 034040
005714' 040440 042116 030440
005722' 020062 042523 047503
005730' 042116 022523 000116
005736' 040445 042522 044503 FORM23::.ASCIZ /%ARECIEVED INTERRUPT AT %D1%A SECONDS%N/
005744' 053105 042105 044440
005752' 052116 051105 052522
005760' 052120 040440 020124
005766' 042045 022461 020101
005774' 042523 047503 042116
006002' 022523 000116
006006' 040445 044515 051103 FORM24::.ASCIZ /%AMICRO TEST %D2%A HUNG%N/
006014' 020117 042524 052123
006022' 022440 031104 040445
006030' 044040 047125 022507
006036' 000116
006040' 040445 040504 040524 FORM25::.ASCIZ /%ADATA WRITTEN TO 'DMATO' REGISTER = %O6%N/
006046' 053440 044522 052124
006054' 047105 052040 020117
006062' 042047 040515 047524
006070' 020047 042522 044507
006076' 052123 051105 036440
006104' 022440 033117 047045
006112'    000
006113'    045 042101 052101 FORM26::.ASCIZ /%ADATA READ FROM 'DMATO' REGISTER  = %O6%N/
006120' 020101 042522 042101
006126' 043040 047522 020115
006134' 042047 040515 047524
006142' 020047 042522 044507
006150' 052123 051105 020040
006156' 020075 047445 022466
006164' 000116
006166' 040445 040504 040524 FORM27::.ASCIZ /%ADATA SHOULD BE: %O6%A DATA WAS: %O6%N/
006174' 051440 047510 046125
006202' 020104 042502 020072
006210' 047445 022466 020101
006216' 040504 040524 053440
006224' 051501 020072 047445
006232' 022466 000116
```

```
006236' 052045  047045     000  FORM28::.ASCIZ  /%T%N/
006243'    045  040501  042104  FORM29::.ASCIZ  /%AADDRESS = %06%S2/
006250' 042522  051523  036440
006256' 022440  033117  051445
006264' 000062
006266' 040445  042522  042503  FORM30::.ASCIZ  /%ARECEIVER STATUS:%06%N/
006274' 053111  051105  051440
006302' 040524  052524  035123
006310' 047445  022466  000116
006316' 040445  052502  043106  FORM31::.ASCIZ  /%ABUFFER OFFSET:%06%N/
006324' 051105  047440  043106
006332' 042523  035124  047445
006340' 022466  000116
006344' 040445  051124  047101  FORM32::.ASCIZ  /%ATRANSMIT STATUS WORD %D1%S1%T%A NOT CLEAR AFTER TRANSMIT%N/
006352' 046523  052111  051440
006360' 040524  052524  020123
006366' 047527  042122  022440
006374' 030504  051445  022461
006402' 022524  020101  047516
006410' 020124  046103  040505
006416' 020122  043101  042524
006424' 020122  051124  047101
006432' 046523  052111  047045
006440'    000
006441'    045  052101  040522  FORM33::.ASCIZ  /%ATRANSMIT BYTE COUNT = %D4%N/
006446' 051516  044515  020124
006454' 054502  042524  041440
006462' 052517  052116  036440
006470' 022440  032104  047045
006476'    000
006477'    045  051101  041505  FORM34::.ASCIZ  /%ARECEIVE BYTE COUNT SHOULD BE: %D4%S3%AWAS: %D4%N/
006504' 044505  042526  041040
006512' 052131  020105  047503
006520' 047125  020124  044123
006526' 052517  042114  041040
006534' 035105  022440  032104
006542' 051445  022463  053501
006550' 051501  020072  042045
006556' 022464  000116
006562' 040445  041501  052524  FORM35::.ASCIZ  /%AACTUAL NUMBER OF BYTES RECEIVED SHOULD BE: %D4%A   WAS: %D4%N/
006570' 046101  047040  046525
006576' 042502  020122  043117
006604' 041040  052131  051505
006612' 051040  041505  044505
006620' 042526  020104  044123
006626' 052517  042114  041040
006634' 035105  022440  032104
006642' 040445  020040  053440
006650' 051501  020072  042045
006656' 022464  000116
006662' 040445  044514  045516  FORM36::.ASCIZ  /%ALINK MEMORY ADDRESS = %06%N/
006670' 046440  046505  051117
006676' 020131  042101  051104
006704' 051505  020123  020075
006712' 047445  022466  000116
006720' 040445  051124  047101  FORM37::.ASCII  /%ATRANSMIT STATUS INFORMATION INCORRECT AFTER LOOPBACK STEP /
```

```
2562  006726' 046523  052111  051440
2563  006734' 040524  052524  020123
2564  006742' 047111  047506  046522
2565  006750' 052101  047511  020116
2566  006756' 047111  047503  051122
2567  006764' 041505  020124  043101
2568  006772' 042524  020122  047514
2569  007000' 050117  040502  045503
2570  007006' 051440  042524  020120
2571  007014' 052516  041115  051105          .ASCIZ  /NUMBER %D2%N/
2572  007022' 022440  031104  047045
2573  007030'    000
2574  007031'    045  052101  040522  FORM38::.ASCIZ  /%ATRANSMIT STATUS WORD %D1%A SHOULD BE: %O6%A WAS: %O6%N/
2575  007036' 051516  044515  020124
2576  007044' 052123  052101  051525
2577  007052' 053440  051117  020104
2578  007060' 042045  022461  020101
2579  007066' 044123  052517  042114
2580  007074' 041040  035105  022440
2581  007102' 033117  040445  053440
2582  007110' 051501  020072  047445
2583  007116' 022466  000116
2584  007122' 040445  042124  020122  FORM39::.ASCIZ  /%ATDR COUNTER NOT INCREMENTING%N/
2585  007130' 047503  047125  042524
2586  007136' 020122  047516  020124
2587  007144' 047111  051103  046505
2588  007152' 047105  044524  043516
2589  007160' 047045     000
2590  007163'    045  051101  052105  FORM40::.ASCIZ  /%ARETRY BACKOFF WAIT TIME INTERVAL NOT VARYING%N%N/
2591  007170' 054522  041040  041501
2592  007176' 047513  043106  053440
2593  007204' 044501  020124  044524
2594  007212' 042515  044440  052116
2595  007220' 051105  040526  020114
2596  007226' 047516  020124  040526
2597  007234' 054522  047111  022507
2598  007242' 022516  000116
2599  007246' 040445  042522  051124  FORM41::.ASCIZ  /%ARETRY #%S4%AWAIT INTERVAL COUNT%N%N/
2600  007254' 020131  022443  032123
2601  007262' 040445  040527  052111
2602  007270' 044440  052116  051105
2603  007276' 040526  020114  047503
2604  007304' 047125  022524  022516
2605  007312' 000116
2606  007314' 042045  022465  033523  FORM42::.ASCIZ  /%D5%S7/
2607  007322'    000
2608  007323'    045  032504  047045  FORM43::.ASCIZ  /%D5%N/
2609  007330'    000
2610  007331'    045  052101  042510  FORM44::.ASCIZ  /%ATHE 48 BIT DESTINATION ADDRESS PATTERN IS:%N/
2611  007336' 032040  020070  044502
2612  007344' 020124  042504  052123
2613  007352' 047111  052101  047511
2614  007360' 020116  042101  051104
2615  007366' 051505  020123  040520
2616  007374' 052124  051105  020116
2617  007402' 051511  022472  000116
```

```
007410' 040445  047514  042527  FORM45::.ASCIZ  /%ALOWER ORDER = %O6%N/
007416' 020122  051117  042504
007424' 020122  020075  047445
007432' 022466  000116
007436' 040445  044515  042104  FORM46::.ASCIZ  /%AMIDDLE ORDER = %O6%N/
007444' 042514  047440  042122
007452' 051105  036440  022440
007460' 033117  047045     000
007465'    045  052501  050120  FORM47::.ASCIZ  /%AUPPER ORDER  = %O6%N/
007472' 051105  047440  042122
007500' 051105  020040  020075
007506' 047445  022466  000116
007514' 040445  042504  047125  FORM48::.ASCIZ  /%ADEUNA FAILED TO RECOGNIZE A STATION ADDRESS THAT MATCHES%N/
007522' 020101  040506  046111
007530' 042105  052040  020117
007536' 042522  047503  047107
007544' 055111  020105  020101
007552' 052123  052101  047511
007560' 020116  042101  051104
007566' 051505  020123  044124
007574' 052101  046440  052101
007602' 044103  051505  047045
007610'    000
007611'    045  052101  042510  FORM49::.ASCIZ  /%ATHE STATION ADDRESS IN RAM POSITION %D2%N/
007616' 051440  040524  044524
007624' 047117  040440  042104
007632' 042522  051523  044440
007640' 020116  040522  020115
007646' 047520  044523  044524
007654' 047117  022440  031104
007662' 047045     000
007665'    045  051101  041505  FORM50::.ASCIZ  /%ARECEIVER STATUS WORD 0 SHOULD BE: %O6%S3%AWAS: %O6%N/
007672' 044505  042526  020122
007700' 052123  052101  051525
007706' 053440  051117  020104
007714' 020060  044123  052517
007722' 042114  041040  035105
007730' 022440  033117  051445
007736' 022463  053501  051501
007744' 020072  047445  022466
007752' 000116
007754' 040445  051103  020103  FORM51::.ASCIZ  /%ACRC ERROR BIT NOT SET%N/
007762' 051105  047522  020122
007770' 044502  020124  047516
007776' 020124  042523  022524
010004' 000116
010006' 040445  051105  047522  FORM52::.ASCIZ  /%AERROR SUMMARY BIT NOT SET%N/
010014' 020122  052523  046515
010022' 051101  020131  044502
010030' 020124  047516  020124
010036' 042523  022524  000116
010044' 040445  052516  041115  FORM53::.ASCIZ  /%ANUMBER OF DATA BYTES TRANSMITTED: %D4%N/
010052' 051105  047440  020106
010060' 040504  040524  041040
010066' 052131  051505  052040
010074' 040522  051516  044515
```

```
010102' 052124 042105 020072
010110' 042045 022464 000116
010116' 040445 040504 040524    FORM54::.ASCIZ  /%ADATA PATTERN: %O6%N/
010124' 050040 052101 042524
010132' 047122 020072 047445
010140' 022466 000116
010144' 040445 042504 047125    FORM55::.ASCIZ  /%ADEUNA FAILED TO REJECT A RUNT PACKET%N/
010152' 020101 040506 046111
010160' 042105 052040 020117
010166' 042522 042512 052103
010174' 040440 051040 047125
010202' 020124 040520 045503
010210' 052105 047045    000
010215'    045 042101 052505    FORM56::.ASCIZ  /%ADEUNA FAILED TO RECOVER RECEIVE BUFFER%N/
010222' 040516 043040 044501
010230' 042514 020104 047524
010236' 051040 041505 053117
010244' 051105 051040 041505
010252' 044505 042526 041040
010260' 043125 042506 022522
010266' 000116
010270' 040445 052516 041115    FORM57::.ASCIZ  /%ANUMBER OF BYTES IN RUNT PACKET: %D2%N/
010276' 051105 047440 020106
010304' 054502 042524 020123
010312' 047111 051040 047125
010320' 020124 040520 045503
010326' 052105 020072 042045
010334' 022462 000116
010340' 040445 052516 041115    FORM58::.ASCIZ  /%ANUMBER OF BYTES IN LEGITIMATE PACKET: %D4%N/
010346' 051105 047440 020106
010354' 054502 042524 020123
010362' 047111 046040 043505
010370' 052111 046511 052101
010376' 020105 040520 045503
010404' 052105 020072 042045
010412' 022464 000116
010416' 040445 042522 042503    FORM59::.ASCIZ  /%ARECEIVE BUFFER ADDRESS AVAILABLE BEFORE RECEPTION: %O6%N/
010424' 053111 020105 052502
010432' 043106 051105 040440
010440' 042104 042522 051523
010446' 040440 040526 046111
010454' 041101 042514 041040
010462' 043105 051117 020105
010470' 042522 042503 052120
010476' 047511 035116 022440
010504' 033117 047045    000
010511'    045 051101 041505    FORM60::.ASCIZ  /%ARECEIVE BUFFER ADDRESS COMPLETED AFTER RECEPTION%N/
010516' 044505 042526 041040
010524' 043125 042506 020122
010532' 042101 051104 051505
010540' 020123 047503 050115
010546' 042514 042524 020104
010554' 043101 042524 020122
010562' 042522 042503 052120
010570' 047511 022516 000116
010576' 040445 040504 040524    FORM61::.ASCIZ  /%ADATA COMPARE ERROR IN RECOVERED RECEIVE BUFFER%N/
```

```
010604' 041440  046517  040520
010612' 042522  042440  051122
010620' 051117  044440  020116
010626' 042522  047503  042526
010634' 042522  020104  042522
010642' 042503  053111  020105
010650' 052502  043106  051105
010656' 047045     000
010661'    045  022524  020101  FORM62::.ASCIZ  /%T%A DID NOT CLEAR AFTER WRITING 1 TO IT%N/
010666' 044504  020104  047516
010674' 020124  046103  040505
010702' 020122  043101  042524
010710' 020122  051127  052111
010716' 047111  020107  020061
010724' 047524  044440  022524
010732' 000116
010734' 040445  042523  043114  FORM63::.ASCIZ  /%ASELF TEST ERROR CODE = %O2%N/
010742' 052040  051505  020124
010750' 051105  047522  020122
010756' 047503  042504  036440
010764' 022440  031117  047045
010772'    000
010773'    045  022516  051101  FORM64::.ASCIZ  /%N%AROM MICROCODE VERSION (DECIMAL): %D2/
011000' 046517  046440  041511
011006' 047522  047503  042504
011014' 053040  051105  044523
011022' 047117  024040  042504
011030' 044503  040515  024514
011036' 020072  042045  000062
011044' 047045  040445  053523  FORM65::.ASCIZ  /%N%ASWITCH PACK = %O6/
011052' 052111  044103  050040
011060' 041501  020113  020075
011066' 047445  000066
011072' 040445  047516  044440  FORM66::.ASCIZ  /%ANO INTERRUPT FROM TRANSMIT STATE MACHINE TO T-11%N/
011100' 052116  051105  052522
011106' 052120  043040  047522
011114' 020115  051124  047101
011122' 046523  052111  051440
011130' 040524  042524  046440
011136' 041501  044510  042516
011144' 052040  020117  026524
011152' 030461  047045     000
011157'    045  047101  020117  FORM67::.ASCIZ  /%ANO INTERRUPT FROM RECEIVE STATE MACHINE TO T-11%N/
011164' 047111  042524  051122
011172' 050125  020124  051106
011200' 046517  051040  041505
011206' 044505  042526  051440
011214' 040524  042524  046440
011222' 041501  044510  042516
011230' 052040  020117  026524
011236' 030461  047045     000
011243'    045  051101  041505  FORM68::.ASCIZ  /%ARECEIVER BYTE COUNTER FAILED TO LOCK UP AT MAXIMUM VALUE%N/
011250' 044505  042526  020122
011256' 054502  042524  041440
011264' 052517  052116  051105
011272' 043040  044501  042514
```

```
011300' 020104  047524  046040
011306' 041517  020113  050125
011314' 040440  020124  040515
011322' 044530  052515  020115
011330' 040526  052514  022505
011336' 000116
011340' 040445  042504  047125  FORM69::.ASCII  /%ADEUNA DID NOT REJECT A PACKET TRANSMITTED TO ITSELF/
011346' 020101  044504  020104
011354' 047516  020124  042522
011362' 042512  052103  040440
011370' 050040  041501  042513
011376' 020124  051124  047101
011404' 046523  052111  042524
011412' 020104  047524  044440
011420' 051524  046105     106
011425'    045  020101  047111          .ASCIZ  /%A IN HALF-DUPLEX MODE%N/
011432' 044040  046101  026506
011440' 052504  046120  054105
011446' 046440  042117  022505
011454' 000116
011456' 040445  051124  047101  FORM70::.ASCIZ  /%ATRANSMIT BUFFER ADDRESS = %06%N/
011464' 046523  052111  041040
011472' 043125  042506  020122
011500' 042101  051104  051505
011506' 020123  020075  047445
011514' 022466  000116
011520' 040445  042522  042503  FORM71::.ASCIZ  /%ARECEIVE BUFFER ADDRESS  = %06%N/
011526' 053111  020105  052502
011534' 043106  051105  040440
011542' 042104  042522  051523
011550' 020040  020075  047445
011556' 022466  000116
011562' 040445  030524  020061  FORM72::.ASCIZ  /%AT11 LINK MEMORY PARITY ERROR OCCURRED%N/
011570' 044514  045516  046440
011576' 046505  051117  020131
011604' 040520  044522  054524
011612' 042440  051122  051117
011620' 047440  041503  051125
011626' 042522  022504  000116
011634' 040445  030524  020061  FORM73::.ASCIZ  /%AT11 NPR TIMEOUT ERROR OCCURRED%N/
011642' 050116  020122  044524
011650' 042515  052517  020124
011656' 051105  047522  020122
011664' 041517  052503  051122
011672' 042105  047045     000
011677'    045  052101  030461  FORM74::.ASCIZ  /%AT11 NON-EXISTANT MEMORY TIMEOUT OCCURRED%N/
011704' 047040  047117  042455
011712' 044530  052123  047101
011720' 020124  042515  047515
011726' 054522  052040  046511
011734' 047505  052125  047440
011742' 041503  051125  042522
011750' 022504  000116
011754' 040445  030524  020061  FORM75::.ASCIZ  /%AT11 UNEXPECTED INTERRUPT OCCURRED%N/
011762' 047125  054105  042520
011770' 052103  042105  044440
```

```
011776' 052116  051105  052522
012004' 052120  047440  041503
012012' 051125  042522  022504
012020' 000116
012022' 040445  040515  041524  FORM76::.ASCIZ  /%AMATCH BIT FAILED TO SET%N/
012030' 020110  044502  020124
012036' 040506  046111  042105
012044' 052040  020117  042523
012052' 022524  000116
012056' 040445  040515  041524  FORM77::.ASCIZ  /%AMATCH BIT SET BUT NO RECEIVER INTERRUPT%N/
012064' 020110  044502  020124
012072' 042523  020124  052502
012100' 020124  047516  051040
012106' 041505  044505  042526
012114' 020122  047111  042524
012122' 051122  050125  022524
012130' 000116
012132' 040445  044123  052517  FORM78::.ASCIZ  /%ASHOULD BE: %O6%S2%AWAS: %O6%N/
012140' 042114  041040  035105
012146' 022440  033117  051445
012154' 022462  053501  051501
012162' 020072  047445  022466
012170' 000116
012172' 040445  042504  047125  FORM79::.ASCIZ  /%ADEUNA FAILED TO REJECT A DATAGRAM.%N/
012200' 020101  040506  046111
012206' 042105  052040  020117
012214' 042522  042512  052103
012222' 040440  042040  052101
012230' 043501  040522  027115
012236' 047045     000
012241'    045  042101  051505  FORM80::.ASCIZ  /%ADESTINATION ADDRESS OF DATAGRAM IS ALL 1'S%N/
012246' 044524  040516  044524
012254' 047117  040440  042104
012262' 042522  051523  047440
012270' 020106  040504  040524
012276' 051107  046501  044440
012304' 020123  046101  020114
012312' 023461  022523  000116
012320' 040445  052123  052101  FORM81::.ASCIZ  /%ASTATION ADDRESS RAM IS ALL 0'S%N/
012326' 047511  020116  042101
012334' 051104  051505  020123
012342' 040522  020115  051511
012350' 040440  046114  030040
012356' 051447  047045     000
012363'    045  052101  046511  FORM82::.ASCIZ  /%ATIMEOUT WAITING FOR MICROCODE TO ENTER MICROMONITOR%N/
012370' 047505  052125  053440
012376' 044501  044524  043516
012404' 043040  051117  046440
012412' 041511  047522  047503
012420' 042504  052040  020117
012426' 047105  042524  020122
012434' 044515  051103  046517
012442' 047117  052111  051117
012450' 047045     000
012453'    045  050101  051503  FORM83::.ASCIZ  /%APCSR%D1%A = %O6%N/                  :MAC001
012460' 022522  030504  040445
```

```
012466' 036440  022440  033117
012474' 047045     000
        012500'                .EVEN
```

```
                                .SBTTL  GLOBAL ERROR REPORT SECTION

                                ;++
                                ; THE GLOBAL ERROR REPORT SECTION CONTAINS MESSAGE PRINTING AREAS
                                ; USED BY MORE THAN TEST TO OUTPUT ADDITIONAL ERROR INFORMATION.  PRINTB
                                ; (BASIC) AND PRINTX (EXTENDED) CALLS ARE USED TO CALL PRINT SERVICES.
                                ;--




012500'                         BGNMSG  RACMG1
012500'                                                                 RACMG1::
012500'                                 PRINTB  #FORM1,CSRNUM
012500' 013746  000304'                                                         MOV     CSRNUM,-(SP)
012504' 012746  004141'                                                         MOV     #FORM1,-(SP)
012510' 012746  000002                                                          MOV     #2,-(SP)
012514' 010600                                                                  MOV     SP,R0
012516' 104414                                                                  TRAP    C$PNTB
012520' 062706  000006                                                          ADD     #6,SP
012524'                         ENDMSG
012524'                                                                 L10001:
012524' 104423                                                                  TRAP    C$MSG

012526'                         BGNMSG  RACMG2
012526'                                                                 RACMG2::
012526'                                 PRINTB  #FORM2,CSRNUM,BITNUM,BITSTA
012526' 013746  000312'                                                         MOV     BITSTA,-(SP)
012532' 013746  000306'                                                         MOV     BITNUM,-(SP)
012536' 013746  000304'                                                         MOV     CSRNUM,-(SP)
012542' 012746  004176'                                                         MOV     #FORM2,-(SP)
012546' 012746  000004                                                          MOV     #4,-(SP)
012552' 010600                                                                  MOV     SP,R0
012554' 104414                                                                  TRAP    C$PNTB
012556' 062706  000012                                                          ADD     #12,SP
012562' 004737  017764'                 JSR     PC,PRNPCR               ;PRINT PCSR'S           ;MAC001
012566'                         ENDMSG
012566'                                                                 L10002:
012566' 104423                                                                  TRAP    C$MSG

012570'                         BGNMSG  RACMG3
012570'                                                                 RACMG3::
012570'                                 PRINTB  #FORM3,CSRNUM
012570' 013746  000304'                                                         MOV     CSRNUM,-(SP)
012574' 012746  004234'                                                         MOV     #FORM3,-(SP)
012600' 012746  000002                                                          MOV     #2,-(SP)
012604' 010600                                                                  MOV     SP,R0
012606' 104414                                                                  TRAP    C$PNTB
012610' 062706  000006                                                          ADD     #6,SP
012614'                                 PRINTB  #FORM4,R3,R4
012614' 010446                                                                  MOV     R4,-(SP)
012616' 010346                                                                  MOV     R3,-(SP)
012620' 012746  004303'                                                         MOV     #FORM4,-(SP)
012624' 012746  000003                                                          MOV     #3,-(SP)
012630' 010600                                                                  MOV     SP,R0
012632' 104414                                                                  TRAP    C$PNTB
```

```
012634' 062706  000010                                          ADD     #10,SP
012640'                 ENDMSG
012640'                                                 L10003:
012640' 104423                                                  TRAP    C$MSG

012642'                 BGNMSG  RACMG4
012642'                                                 RACMG4::
012642'                         PRINTB  #FORM5
012642' 012746  004352'                                         MOV     #FORM5,-(SP)
012646' 012746  000001                                          MOV     #1,-(SP)
012652' 010600                                                  MOV     SP,R0
012654' 104414                                                  TRAP    C$PNTB
012656' 062706  000004                                          ADD     #4,SP
012662' 004737  017764'         JSR     PC,PRNPCR       ;PRINT PCSR'S   ;MAC001
012666'                 ENDMSG
012666'                                                 L10004:
012666' 104423                                                  TRAP    C$MSG

012670'                 BGNMSG  RACMG7
012670'                                                 RACMG7::
012670'                         PRINTB  #FORM8
012670' 012746  004555'                                         MOV     #FORM8,-(SP)
012674' 012746  000001                                          MOV     #1,-(SP)
012700' 010600                                                  MOV     SP,R0
012702' 104414                                                  TRAP    C$PNTB
012704' 062706  000004                                          ADD     #4,SP
012710' 004737  017764'         JSR     PC,PRNPCR       ;PRINT PCSR'S   ;MAC001
012714'                 ENDMSG
012714'                                                 L10005:
012714' 104423                                                  TRAP    C$MSG

012716'                 BGNMSG  MSG1
012716'                                                 MSG1::
012716'                         PRINTB  #FORM6,BITNAM,BITSTA,PWHEN,PCOMND
012716' 013746  000316'                                         MOV     PCOMND,-(SP)
012722' 013746  000314'                                         MOV     PWHEN,-(SP)
012726' 013746  000312'                                         MOV     BITSTA,-(SP)
012732' 013746  000310'                                         MOV     BITNAM,-(SP)
012736' 012746  004415'                                         MOV     #FORM6,-(SP)
012742' 012746  000005                                          MOV     #5,-(SP)
012746' 010600                                                  MOV     SP,R0
012750' 104414                                                  TRAP    C$PNTB
012752' 062706  000014                                          ADD     #14,SP
012756' 004737  017764'         JSR     PC,PRNPCR       ;PRINT PCSR'S   ;MAC001
012762'                 ENDMSG
012762'                                                 L10006:
012762' 104423                                                  TRAP    C$MSG

012764'                 BGNMSG  MSG2
012764'                                                 MSG2::
012764'                         PRINTB  #FORM7
012764' 012746  004473'                                         MOV     #FORM7,-(SP)
012770' 012746  000001                                          MOV     #1,-(SP)
012774' 010600                                                  MOV     SP,R0
012776' 104414                                                  TRAP    C$PNTB
013000' 062706  000004                                          ADD     #4,SP
```

```
013004'                           PRINTB  #FORM13,R4
013004' 010446                                                        MOV     R4,-(SP)
013006' 012746  005123'                                               MOV     #FORM13,-(SP)
013012' 012746  000002                                                MOV     #2,-(SP)
013016' 010600                                                        MOV     SP,R0
013020' 104414                                                        TRAP    C$PNTB
013022' 062706  000006                                                ADD     #6,SP
013026'                   ENDMSG
013026'                                                       L10007:
013026' 104423                                                        TRAP    C$MSG

013030'                   BGNMSG  MSG3
013030'                                                       MSG3::
013030'                           PRINTB  #FORM9,CSRNUM,BITNAM,BITSTA,PWHEN
013030' 013746  000314'                                               MOV     PWHEN,-(SP)
013034' 013746  000312'                                               MOV     BITSTA,-(SP)
013040' 013746  000310'                                               MOV     BITNAM,-(SP)
013044' 013746  000304'                                               MOV     CSRNUM,-(SP)
013050' 012746  004637'                                               MOV     #FORM9,-(SP)
013054' 012746  000005                                                MOV     #5,-(SP)
013060' 010600                                                        MOV     SP,R0
013062' 104414                                                        TRAP    C$PNTB
013064' 062706  000014                                                ADD     #14,SP
013070' 004737  017764'           JSR     PC,PRNPCR           ;PRINT PCSR'S       ;MAC001
013074'                   ENDMSG
013074'                                                       L10010:
013074' 104423                                                        TRAP    C$MSG

013076'                   BGNMSG  MSG4
013076'                                                       MSG4::
013076'                           PRINTB  #FORM10
013076' 012746  004715'                                               MOV     #FORM10,-(SP)
013102' 012746  000001                                                MOV     #1,-(SP)
013106' 010600                                                        MOV     SP,R0
013110' 104414                                                        TRAP    C$PNTB
013112' 062706  000004                                                ADD     #4,SP
013116' 004737  017764'           JSR     PC,PRNPCR           ;PRINT PCSR'S       ;MAC001
013122'                   ENDMSG
013122'                                                       L10011:
013122' 104423                                                        TRAP    C$MSG

013124'                   BGNMSG  MSG5
013124'                                                       MSG5::
013124'                           PRINTB  #FORM11
013124' 012746  005001'                                               MOV     #FORM11,-(SP)
013130' 012746  000001                                                MOV     #1,-(SP)
013134' 010600                                                        MOV     SP,R0
013136' 104414                                                        TRAP    C$PNTB
013140' 062706  000004                                                ADD     #4,SP
013144'                           PRINTB  #FORM12,R1,(R4),(R1)
013144' 011146                                                        MOV     (R1),-(SP)
013146' 011446                                                        MOV     (R4),-(SP)
013150' 010146                                                        MOV     R1,-(SP)
013152' 012746  005030'                                               MOV     #FORM12,-(SP)
013156' 012746  000004                                                MOV     #4,-(SP)
013162' 010600                                                        MOV     SP,R0
```

```
013164' 104414                                                 TRAP    C$PNTB
013166' 062706  000012                                         ADD     #12,SP
013172'                 ENDMSG
013172'                                                L10012:
013172' 104423                                                 TRAP    C$MSG

013174'                 BGNMSG  MSG6
013174'                                                MSG6::
013174'                         PRINTB  #FORM11
013174' 012746  005001'                                        MOV     #FORM11,-(SP)
013200' 012746  000001                                         MOV     #1,-(SP)
013204' 010600                                                 MOV     SP,R0
013206' 104414                                                 TRAP    C$PNTB
013210' 062706  000004                                         ADD     #4,SP
013214'                         PRINTB  #FORM15,R1
013214' 010146                                                 MOV     R1,-(SP)
013216' 012746  005174'                                        MOV     #FORM15,-(SP)
013222' 012746  000002                                         MOV     #2,-(SP)
013226' 010600                                                 MOV     SP,R0
013230' 104414                                                 TRAP    C$PNTB
013232' 062706  000006                                         ADD     #6,SP
013236'                         PRINTB  #FORM16,R2
013236' 010246                                                 MOV     R2,-(SP)
013240' 012746  005250'                                        MOV     #FORM16,-(SP)
013244' 012746  000002                                         MOV     #2,-(SP)
013250' 010600                                                 MOV     SP,R0
013252' 104414                                                 TRAP    C$PNTB
013254' 062706  000006                                         ADD     #6,SP
013260'                 ENDMSG
013260'                                                L10013:
013260' 104423                                                 TRAP    C$MSG

013262'                 BGNMSG  MSG7
013262'                                                MSG7::
013262'                         PRINTB  #FORM17
013262' 012746  005326'                                        MOV     #FORM17,-(SP)
013266' 012746  000001                                         MOV     #1,-(SP)
013272' 010600                                                 MOV     SP,R0
013274' 104414                                                 TRAP    C$PNTB
013276' 062706  000004                                         ADD     #4,SP
013302' 004737  017764'         JSR     PC,PRNPCR                   ;PRINT PCSR'S      ;MAC001
013306'                 ENDMSG
013306'                                                L10014:
013306' 104423                                                 TRAP    C$MSG

013310'                 BGNMSG  MSG8
013310'                                                MSG8::
013310'                         PRINTB  #FORM18
013310' 012746  005411'                                        MOV     #FORM18,-(SP)
013314' 012746  000001                                         MOV     #1,-(SP)
013320' 010600                                                 MOV     SP,R0
013322' 104414                                                 TRAP    C$PNTB
013324' 062706  000004                                         ADD     #4,SP
013330'                 ENDMSG
013330'                                                L10015:
013330' 104423                                                 TRAP    C$MSG
```

```

013332'                        BGNMSG  MSG9
013332'                                                MSG9::
013332'                                PRINTB  #FORM19,@BITNAM
013332' 017746  164752                                          MOV     @BITNAM,-(SP)
013336' 012746  005467'                                         MOV     #FORM19,-(SP)
013342' 012746  000002                                          MOV     #2,-(SP)
013346' 010600                                                  MOV     SP,R0
013350' 104414                                                  TRAP    C$PNTB
013352' 062706  000006                                          ADD     #6,SP
013356'                        ENDMSG
013356'                                                L10016:
013356' 104423                                                  TRAP    C$MSG

013360'                        BGNMSG  MSG10
013360'                                                MSG10::
013360'                                PRINTB  #FORM20
013360' 012746  005544'                                         MOV     #FORM20,-(SP)
013364' 012746  000001                                          MOV     #1,-(SP)
013370' 010600                                                  MOV     SP,R0
013372' 104414                                                  TRAP    C$PNTB
013374' 062706  000004                                          ADD     #4,SP
013400'                        ENDMSG
013400'                                                L10017:
013400' 104423                                                  TRAP    C$MSG

013402'                        BGNMSG  MSG11
013402'                                                MSG11::
013402'                                PRINTB  #FORM21
013402' 012746  005604'                                         MOV     #FORM21,-(SP)
013406' 012746  000001                                          MOV     #1,-(SP)
013412' 010600                                                  MOV     SP,R0
013414' 104414                                                  TRAP    C$PNTB
013416' 062706  000004                                          ADD     #4,SP
013422'                                PRINTB  #FORM22
013422' 012746  005656'                                         MOV     #FORM22,-(SP)
013426' 012746  000001                                          MOV     #1,-(SP)
013432' 010600                                                  MOV     SP,R0
013434' 104414                                                  TRAP    C$PNTB
013436' 062706  000004                                          ADD     #4,SP
013442'                                PRINTB  #FORM23,R1
013442' 010146                                                  MOV     R1,-(SP)
013444' 012746  005736'                                         MOV     #FORM23,-(SP)
013450' 012746  000002                                          MOV     #2,-(SP)
013454' 010600                                                  MOV     SP,R0
013456' 104414                                                  TRAP    C$PNTB
013460' 062706  000006                                          ADD     #6,SP
013464'                        ENDMSG
013464'                                                L10020:
013464' 104423                                                  TRAP    C$MSG

013466'                        BGNMSG  MSG12
013466'                                                MSG12::
013466'                                PRINTB  #FORM24,R2
013466' 010246                                                  MOV     R2,-(SP)
013470' 012746  006006'                                         MOV     #FORM24,-(SP)
```

```
013474' 012746  000002                                           MOV     #2,-(SP)
013500' 010600                                                   MOV     SP,R0
013502' 104414                                                   TRAP    C$PNTB
013504' 062706  000006                                           ADD     #6,SP
013510'                 ENDMSG
013510'                                                 L10021:
013510' 104423                                                   TRAP    C$MSG

013512'                 BGNMSG  MSG13
013512'                                                 MSG13::
013512'                         PRINTB  #FORM25,R3
013512' 010346                                                   MOV     R3,-(SP)
013514' 012746  006040'                                          MOV     #FORM25,-(SP)
013520' 012746  000002                                           MOV     #2,-(SP)
013524' 010600                                                   MOV     SP,R0
013526' 104414                                                   TRAP    C$PNTB
013530' 062706  000006                                           ADD     #6,SP
013534'                         PRINTB  #FORM26,R4
013534' 010446                                                   MOV     R4,-(SP)
013536' 012746  006113'                                          MOV     #FORM26,-(SP)
013542' 012746  000002                                           MOV     #2,-(SP)
013546' 010600                                                   MOV     SP,R0
013550' 104414                                                   TRAP    C$PNTB
013552' 062706  000006                                           ADD     #6,SP
013556'                 ENDMSG
013556'                                                 L10022:
013556' 104423                                                   TRAP    C$MSG

013560'                 BGNMSG  MSG14
013560'                                                 MSG14::
013560'                         PRINTB  #FORM27,R3,R4
013560' 010446                                                   MOV     R4,-(SP)
013562' 010346                                                   MOV     R3,-(SP)
013564' 012746  006166'                                          MOV     #FORM27,-(SP)
013570' 012746  000003                                           MOV     #3,-(SP)
013574' 010600                                                   MOV     SP,R0
013576' 104414                                                   TRAP    C$PNTB
013600' 062706  000010                                           ADD     #10,SP
013604'                 ENDMSG
013604'                                                 L10023:
013604' 104423                                                   TRAP    C$MSG

013606'                 BGNMSG  MSG15
013606'                                                 MSG15::
013606'                         PRINTB  #FORM12,R2,(R1),(R2)
013606' 011246                                                   MOV     (R2),-(SP)
013610' 011146                                                   MOV     (R1),-(SP)
013612' 010246                                                   MOV     R2,-(SP)
013614' 012746  005030'                                          MOV     #FORM12,-(SP)
013620' 012746  000004                                           MOV     #4,-(SP)
013624' 010600                                                   MOV     SP,R0
013626' 104414                                                   TRAP    C$PNTB
013630' 062706  000012                                           ADD     #12,SP
013634'                 ENDMSG
013634'                                                 L10024:
013634' 104423                                                   TRAP    C$MSG
```

```
                                BGNMSG  MSG16
013636'                                                          MSG16::
013636'                                 PRINTB  #FORM28,R2
013636' 010246                                                           MOV     R2,-(SP)
013640' 012746  006236'                                                  MOV     #FORM28,-(SP)
013644' 012746  000002                                                   MOV     #2,-(SP)
013650' 010600                                                           MOV     SP,R0
013652' 104414                                                           TRAP    C$PNTB
013654' 062706  000006                                                   ADD     #6,SP
013660'                                 PRINTB  #FORM29,R1
013660' 010146                                                           MOV     R1,-(SP)
013662' 012746  006243'                                                  MOV     #FORM29,-(SP)
013666' 012746  000002                                                   MOV     #2,-(SP)
013672' 010600                                                           MOV     SP,R0
013674' 104414                                                           TRAP    C$PNTB
013676' 062706  000006                                                   ADD     #6,SP
013702'                                 PRINTB  #FORM27,R3,R4
013702' 010446                                                           MOV     R4,-(SP)
013704' 010346                                                           MOV     R3,-(SP)
013706' 012746  006166'                                                  MOV     #FORM27,-(SP)
013712' 012746  000003                                                   MOV     #3,-(SP)
013716' 010600                                                           MOV     SP,R0
013720' 104414                                                           TRAP    C$PNTB
013722' 062706  000010                                                   ADD     #10,SP
013726'                         ENDMSG
013726'                                                          L10025:
013726' 104423                                                           TRAP    C$MSG

013730'                         BGNMSG  MSG17
013730'                                                          MSG17::
013730'                                 PRINTB  #FORM30,R4
013730' 010446                                                           MOV     R4,-(SP)
013732' 012746  006266'                                                  MOV     #FORM30,-(SP)
013736' 012746  000002                                                   MOV     #2,-(SP)
013742' 010600                                                           MOV     SP,R0
013744' 104414                                                           TRAP    C$PNTB
013746' 062706  000006                                                   ADD     #6,SP
013752'                                 PRINTB  #FORM31,R1
013752' 010146                                                           MOV     R1,-(SP)
013754' 012746  006316'                                                  MOV     #FORM31,-(SP)
013760' 012746  000002                                                   MOV     #2,-(SP)
013764' 010600                                                           MOV     SP,R0
013766' 104414                                                           TRAP    C$PNTB
013770' 062706  000006                                                   ADD     #6,SP
013774'                                 PRINTB  #FORM27,R2,R3
013774' 010346                                                           MOV     R3,-(SP)
013776' 010246                                                           MOV     R2,-(SP)
014000' 012746  006166'                                                  MOV     #FORM27,-(SP)
014004' 012746  000003                                                   MOV     #3,-(SP)
014010' 010600                                                           MOV     SP,R0
014012' 104414                                                           TRAP    C$PNTB
014014' 062706  000010                                                   ADD     #10,SP
014020'                         ENDMSG
014020'                                                          L10026:
014020' 104423                                                           TRAP    C$MSG
```

```
        014022'                         BGNMSG  MSG18
        014022'                                                         MSG18::
        014022'                                 PRINTB  #FORM32,R1,BITNAM
        014022' 013746  000310'                                                 MOV     BITNAM,-(SP)
        014026' 010146                                                          MOV     R1,-(SP)
        014030' 012746  006344'                                                 MOV     #FORM32,-(SP)
        014034' 012746  000003                                                  MOV     #3,-(SP)
        014040' 010600                                                          MOV     SP,R0
        014042' 104414                                                          TRAP    C$PNTB
        014044' 062706  000010                                                  ADD     #10,SP
        014050'                         ENDMSG
        014050'                                                         L10027:
        014050' 104423                                                          TRAP    C$MSG

        014052'                         BGNMSG  MSG19
        014052'                                                         MSG19::
        014052'                                 PRINTB  #FORM33,R1
        014052' 010146                                                          MOV     R1,-(SP)
        014054' 012746  006441'                                                 MOV     #FORM33,-(SP)
        014060' 012746  000002                                                  MOV     #2,-(SP)
        014064' 010600                                                          MOV     SP,R0
        014066' 104414                                                          TRAP    C$PNTB
        014070' 062706  000006                                                  ADD     #6,SP
        014074'                                 PRINTB  #FORM34,R2,R3
        014074' 010346                                                          MOV     R3,-(SP)
        014076' 010246                                                          MOV     R2,-(SP)
        014100' 012746  006477'                                                 MOV     #FORM34,-(SP)
        014104' 012746  000003                                                  MOV     #3,-(SP)
        014110' 010600                                                          MOV     SP,R0
        014112' 104414                                                          TRAP    C$PNTB
        014114' 062706  000010                                                  ADD     #10,SP
        014120'                                 PRINTB  #FORM35,R1,R4
        014120' 010446                                                          MOV     R4,-(SP)
        014122' 010146                                                          MOV     R1,-(SP)
        014124' 012746  006562'                                                 MOV     #FORM35,-(SP)
        014130' 012746  000003                                                  MOV     #3,-(SP)
        014134' 010600                                                          MOV     SP,R0
        014136' 104414                                                          TRAP    C$PNTB
        014140' 062706  000010                                                  ADD     #10,SP
        014144'                         ENDMSG
        014144'                                                         L10030:
        014144' 104423                                                          TRAP    C$MSG

        014146'                         BGNMSG  MSG20
        014146'                                                         MSG20::
        014146'                                 PRINTB  #FORM11
        014146' 012746  005001'                                                 MOV     #FORM11,-(SP)
        014152' 012746  000001                                                  MOV     #1,-(SP)
        014156' 010600                                                          MOV     SP,R0
        014160' 104414                                                          TRAP    C$PNTB
        014162' 062706  000004                                                  ADD     #4,SP
        014166'                                 PRINTB  #FORM36,R3
        014166' 010346                                                          MOV     R3,-(SP)
        014170' 012746  006662'                                                 MOV     #FORM36,-(SP)
        014174' 012746  000002                                                  MOV     #2,-(SP)
```

```
014200' 010600                                                  MOV     SP,R0
014202' 104414                                                  TRAP    C$PNTB
014204' 062706 000006                                           ADD     #6,SP
014210'                       PRINTB  #FORM27,R1,R2
014210' 010246                                                  MOV     R2,-(SP)
014212' 010146                                                  MOV     R1,-(SP)
014214' 012746 006166'                                          MOV     #FORM27,-(SP)
014220' 012746 000003                                           MOV     #3,-(SP)
014224' 010600                                                  MOV     SP,R0
014226' 104414                                                  TRAP    C$PNTB
014230' 062706 000010                                           ADD     #10,SP
014234'               ENDMSG
014234'                                                 L10031:
014234' 104423                                                  TRAP    C$MSG

014236'               BGNMSG  MSG21
014236'                                                 MSG21::
014236'                       PRINTB  #FORM11
014236' 012746 005001'                                          MOV     #FORM11,-(SP)
014242' 012746 000001                                           MOV     #1,-(SP)
014246' 010600                                                  MOV     SP,R0
014250' 104414                                                  TRAP    C$PNTB
014252' 062706 000004                                           ADD     #4,SP
014256'                       PRINTB  #FORM29,R1
014256' 010146                                                  MOV     R1,-(SP)
014260' 012746 006243'                                          MOV     #FORM29,-(SP)
014264' 012746 000002                                           MOV     #2,-(SP)
014270' 010600                                                  MOV     SP,R0
014272' 104414                                                  TRAP    C$PNTB
014274' 062706 000006                                           ADD     #6,SP
014300'                       PRINTB  #FORM27,R2,R3
014300' 010346                                                  MOV     R3,-(SP)
014302' 010246                                                  MOV     R2,-(SP)
014304' 012746 006166'                                          MOV     #FORM27,-(SP)
014310' 012746 000003                                           MOV     #3,-(SP)
014314' 010600                                                  MOV     SP,R0
014316' 104414                                                  TRAP    C$PNTB
014320' 062706 000010                                           ADD     #10,SP
014324'               ENDMSG
014324'                                                 L10032:
014324' 104423                                                  TRAP    C$MSG

014326'               BGNMSG  MSG22
014326'                                                 MSG22::
014326' 010146                MOV     R1,-(SP)
014330'                       PRINTB  #FORM37,R2
014330' 010246                                                  MOV     R2,-(SP)
014332' 012746 006720'                                          MOV     #FORM37,-(SP)
014336' 012746 000002                                           MOV     #2,-(SP)
014342' 010600                                                  MOV     SP,R0
014344' 104414                                                  TRAP    C$PNTB
014346' 062706 000006                                           ADD     #6,SP
014352' 005001                CLR     R1              ;TRANSMIT STATUS WORD 0
014354'                       PRINTB  #FORM38,R1,R3,PCBB+2
014354' 013746 000610'                                          MOV     PCBB+2,-(SP)
014360' 010346                                                  MOV     R3,-(SP)
```

```
014362' 010146                                                       MOV     R1,-(SP)
014364' 012746  007031'                                              MOV     #FORM38,-(SP)
014370' 012746  000004                                               MOV     #4,-(SP)
014374' 010600                                                       MOV     SP,R0
014376' 104414                                                       TRAP    C$PNTB
014400' 062706  000012                                               ADD     #12,SP
014404' 005201                  INC     R1              ;TRANSMIT STATUS WORD 1
014406'                         PRINTB  #FORM38,R1,R4,PCBB+4
014406' 013746  000612'                                              MOV     PCBB+4,-(SP)
014412' 010446                                                       MOV     R4,-(SP)
014414' 010146                                                       MOV     R1,-(SP)
014416' 012746  007031'                                              MOV     #FORM38,-(SP)
014422' 012746  000004                                               MOV     #4,-(SP)
014426' 010600                                                       MOV     SP,R0
014430' 104414                                                       TRAP    C$PNTB
014432' 062706  000012                                               ADD     #12,SP
014436' 012601                  MOV     (SP)+,R1
014440'                 ENDMSG
014440'                                                      L10033:
014440' 104423                                                       TRAP    C$MSG

014442'                 BGNMSG  MSG23
014442'                                                      MSG23::
014442'                         PRINTB  #FORM39
014442' 012746  007122'                                              MOV     #FORM39,-(SP)
014446' 012746  000001                                               MOV     #1,-(SP)
014452' 010600                                                       MOV     SP,R0
014454' 104414                                                       TRAP    C$PNTB
014456' 062706  000004                                               ADD     #4,SP
014462'                 ENDMSG
014462'                                                      L10034:
014462' 104423                                                       TRAP    C$MSG

014464'                 BGNMSG  MSG24
014464'                                                      MSG24::
014464'                         PRINTB  #FORM40
014464' 012746  007163'                                              MOV     #FORM40,-(SP)
014470' 012746  000001                                               MOV     #1,-(SP)
014474' 010600                                                       MOV     SP,R0
014476' 104414                                                       TRAP    C$PNTB
014500' 062706  000004                                               ADD     #4,SP
014504'                         PRINTB  #FORM41
014504' 012746  007246'                                              MOV     #FORM41,-(SP)
014510' 012746  000001                                               MOV     #1,-(SP)
014514' 010600                                                       MOV     SP,R0
014516' 104414                                                       TRAP    C$PNTB
014520' 062706  000004                                               ADD     #4,SP
014524' 012701  000001          MOV     #1,R1
014530' 012702  000626'         MOV     #CNTTAB,R2
014534' 012203          10$:    MOV     (R2)+,R3
014536'                         PRINTB  #FORM42,R1
014536' 010146                                                       MOV     R1,-(SP)
014540' 012746  007314'                                              MOV     #FORM42,-(SP)
014544' 012746  000002                                               MOV     #2,-(SP)
014550' 010600                                                       MOV     SP,R0
014552' 104414                                                       TRAP    C$PNTB
```

```
014554' 062706  000006                                          ADD     #6,SP
014560'                         PRINTB  #FORM43,R3
014560' 010346                                                  MOV     R3,-(SP)
014562' 012746  007323'                                         MOV     #FORM43,-(SP)
014566' 012746  000002                                          MOV     #2,-(SP)
014572' 010600                                                  MOV     SP,R0
014574' 104414                                                  TRAP    C$PNTB
014576' 062706  000006                                          ADD     #6,SP
014602' 005201                  INC     R1
014604' 020127  000021          CMP     R1,#17.
014610' 001351                  BNE     10$
014612'                 ENDMSG
014612'                                                 L10035:
014612' 104423                                                  TRAP    C$MSG

014614'                 BGNMSG  MSG25
014614'                                                 MSG25::
014614' 122777  000003  163516  CMPB    #INERR,@PCSR1   ;WAS MATCH BIT SET?
014622' 001011                  BNE     10$             ;YES
014624'                         PRINTB  #FORM76         ;NO
014624' 012746  012022'                                         MOV     #FORM76,-(SP)
014630' 012746  000001                                          MOV     #1,-(SP)
014634' 010600                                                  MOV     SP,R0
014636' 104414                                                  TRAP    C$PNTB
014640' 062706  000004                                          ADD     #4,SP
014644' 000410                  BR      20$
014646'                 10$:    PRINTB  #FORM77
014646' 012746  012056'                                         MOV     #FORM77,-(SP)
014652' 012746  000001                                          MOV     #1,-(SP)
014656' 010600                                                  MOV     SP,R0
014660' 104414                                                  TRAP    C$PNTB
014662' 062706  000004                                          ADD     #4,SP
014666'                 20$:    PRINTB  #FORM44
014666' 012746  007331'                                         MOV     #FORM44,-(SP)
014672' 012746  000001                                          MOV     #1,-(SP)
014676' 010600                                                  MOV     SP,R0
014700' 104414                                                  TRAP    C$PNTB
014702' 062706  000004                                          ADD     #4,SP
014706'                         PRINTB  #FORM45,(R1)
014706' 011146                                                  MOV     (R1),-(SP)
014710' 012746  007410'                                         MOV     #FORM45,-(SP)
014714' 012746  000002                                          MOV     #2,-(SP)
014720' 010600                                                  MOV     SP,R0
014722' 104414                                                  TRAP    C$PNTB
014724' 062706  000006                                          ADD     #6,SP
014730'                         PRINTB  #FORM46,2(R1)
014730' 016146  000002                                          MOV     2(R1),-(SP)
014734' 012746  007436'                                         MOV     #FORM46,-(SP)
014740' 012746  000002                                          MOV     #2,-(SP)
014744' 010600                                                  MOV     SP,R0
014746' 104414                                                  TRAP    C$PNTB
014750' 062706  000006                                          ADD     #6,SP
014754'                         PRINTB  #FORM47,4(R1)
014754' 016146  000004                                          MOV     4(R1),-(SP)
014760' 012746  007465'                                         MOV     #FORM47,-(SP)
014764' 012746  000002                                          MOV     #2,-(SP)
```

```
014770' 010600                                                  MOV     SP,R0
014772' 104414                                                  TRAP    C$PNTB
014774' 062706  000006                                          ADD     #6,SP
015000'                 ENDMSG
015000'                                         L10036:
015000' 104423                                                  TRAP    C$MSG

015002'                 BGNMSG  MSG26
015002'                                         MSG26::
015002'                         PRINTB  #FORM48
015002' 012746  007514'                                         MOV     #FORM48,-(SP)
015006' 012746  000001                                          MOV     #1,-(SP)
015012' 010600                                                  MOV     SP,R0
015014' 104414                                                  TRAP    C$PNTB
015016' 062706  000004                                          ADD     #4,SP
015022'                         PRINTB  #FORM49,R1
015022' 010146                                                  MOV     R1,-(SP)
015024' 012746  007611'                                         MOV     #FORM49,-(SP)
015030' 012746  000002                                          MOV     #2,-(SP)
015034' 010600                                                  MOV     SP,R0
015036' 104414                                                  TRAP    C$PNTB
015040' 062706  000006                                          ADD     #6,SP
015044'                 ENDMSG
015044'                                         L10037:
015044' 104423                                                  TRAP    C$MSG

015046'                 BGNMSG  MSG27
015046'                                         MSG27::
015046'                         PRINTB  #FORM51
015046' 012746  007754'                                         MOV     #FORM51,-(SP)
015052' 012746  000001                                          MOV     #1,-(SP)
015056' 010600                                                  MOV     SP,R0
015060' 104414                                                  TRAP    C$PNTB
015062' 062706  000004                                          ADD     #4,SP
015066'                         PRINTB  #FORM50,R4,R3
015066' 010346                                                  MOV     R3,-(SP)
015070' 010446                                                  MOV     R4,-(SP)
015072' 012746  007665'                                         MOV     #FORM50,-(SP)
015076' 012746  000003                                          MOV     #3,-(SP)
015102' 010600                                                  MOV     SP,R0
015104' 104414                                                  TRAP    C$PNTB
015106' 062706  000010                                          ADD     #10,SP
015112'                 ENDMSG
015112'                                         L10040:
015112' 104423                                                  TRAP    C$MSG

015114'                 BGNMSG  MSG28
015114'                                         MSG28::
015114'                         PRINTB  #FORM52
015114' 012746  010006'                                         MOV     #FORM52,-(SP)
015120' 012746  000001                                          MOV     #1,-(SP)
015124' 010600                                                  MOV     SP,R0
015126' 104414                                                  TRAP    C$PNTB
015130' 062706  000004                                          ADD     #4,SP
015134'                         PRINTB  #FORM50,R4,R3
015134' 010346                                                  MOV     R3,-(SP)
```

```
015136' 010446                                                  MOV     R4,-(SP)
015140' 012746  007665'                                         MOV     #FORM50,-(SP)
015144' 012746  000003                                          MOV     #3,-(SP)
015150' 010600                                                  MOV     SP,R0
015152' 104414                                                  TRAP    C$PNTB
015154' 062706  000010                                          ADD     #10,SP
015160'                 ENDMSG
015160'                                                 L10041:
015160' 104423                                                  TRAP    C$MSG

015162'                 BGNMSG  MSG29
015162'                                                 MSG29::
015162'                         PRINTB  #FORM50,#0,PCBB+4
015162' 013746  000612'                                         MOV     PCBB+4,-(SP)
015166' 012746  000000                                          MOV     #0,-(SP)
015172' 012746  007665'                                         MOV     #FORM50,-(SP)
015176' 012746  000003                                          MOV     #3,-(SP)
015202' 010600                                                  MOV     SP,R0
015204' 104414                                                  TRAP    C$PNTB
015206' 062706  000010                                          ADD     #10,SP
015212'                         PRINTB  #FORM53,PCBB+2
015212' 013746  000610'                                         MOV     PCBB+2,-(SP)
015216' 012746  010044'                                         MOV     #FORM53,-(SP)
015222' 012746  000002                                          MOV     #2,-(SP)
015226' 010600                                                  MOV     SP,R0
015230' 104414                                                  TRAP    C$PNTB
015232' 062706  000006                                          ADD     #6,SP
015236'                         PRINTB  #FORM54,PCBB
015236' 013746  000606'                                         MOV     PCBB,-(SP)
015242' 012746  010116'                                         MOV     #FORM54,-(SP)
015246' 012746  000002                                          MOV     #2,-(SP)
015252' 010600                                                  MOV     SP,R0
015254' 104414                                                  TRAP    C$PNTB
015256' 062706  000006                                          ADD     #6,SP
015262'                 ENDMSG
015262'                                                 L10042:
015262' 104423                                                  TRAP    C$MSG

015264'                 BGNMSG  MSG30
015264'                                                 MSG30::
015264'                         PRINTB  #FORM55
015264' 012746  010144'                                         MOV     #FORM55,-(SP)
015270' 012746  000001                                          MOV     #1,-(SP)
015274' 010600                                                  MOV     SP,R0
015276' 104414                                                  TRAP    C$PNTB
015300' 062706  000004                                          ADD     #4,SP
015304'                         PRINTB  #FORM57,PCBB
015304' 013746  000606'                                         MOV     PCBB,-(SP)
015310' 012746  010270'                                         MOV     #FORM57,-(SP)
015314' 012746  000002                                          MOV     #2,-(SP)
015320' 010600                                                  MOV     SP,R0
015322' 104414                                                  TRAP    C$PNTB
015324' 062706  000006                                          ADD     #6,SP
015330'                 ENDMSG
015330'                                                 L10043:
015330' 104423                                                  TRAP    C$MSG
```

```
     015332'                        BGNMSG  MSG31
     015332'                                                MSG31::
     015332'                                PRINTB  #FORM56
     015332' 012746  010215'                                        MOV     #FORM56,-(SP)
     015336' 012746  000001                                         MOV     #1,-(SP)
     015342' 010600                                                 MOV     SP,R0
     015344' 104414                                                 TRAP    C$PNTB
     015346' 062706  000004                                         ADD     #4,SP
     015352'                                PRINTB  #FORM57,R1
     015352' 010146                                                 MOV     R1,-(SP)
     015354' 012746  010270'                                        MOV     #FORM57,-(SP)
     015360' 012746  000002                                         MOV     #2,-(SP)
     015364' 010600                                                 MOV     SP,R0
     015366' 104414                                                 TRAP    C$PNTB
     015370' 062706  000006                                         ADD     #6,SP
     015374'                                PRINTB  #FORM58,#104.
     015374' 012746  000150                                         MOV     #104.,-(SP)
     015400' 012746  010340'                                        MOV     #FORM58,-(SP)
     015404' 012746  000002                                         MOV     #2,-(SP)
     015410' 010600                                                 MOV     SP,R0
     015412' 104414                                                 TRAP    C$PNTB
     015414' 062706  000006                                         ADD     #6,SP
     015420'                                PRINTB  #FORM59,#100000
     015420' 012746  100000                                         MOV     #100000,-(SP)
     015424' 012746  010416'                                        MOV     #FORM59,-(SP)
     015430' 012746  000002                                         MOV     #2,-(SP)
     015434' 010600                                                 MOV     SP,R0
     015436' 104414                                                 TRAP    C$PNTB
     015440' 062706  000006                                         ADD     #6,SP
     015444'                                PRINTB  #FORM60,#100000,PCBB+2
     015444' 013746  000610'                                        MOV     PCBB+2,-(SP)
     015450' 012746  100000                                         MOV     #100000,-(SP)
     015454' 012746  010511'                                        MOV     #FORM60,-(SP)
     015460' 012746  000003                                         MOV     #3,-(SP)
     015464' 010600                                                 MOV     SP,R0
     015466' 104414                                                 TRAP    C$PNTB
     015470' 062706  000010                                         ADD     #10,SP
     015474'                        ENDMSG
     015474'                                                L10044:
     015474' 104423                                                 TRAP    C$MSG

     015476'                        BGNMSG  MSG32
     015476'                                                MSG32::
     015476'                                PRINTB  #FORM61
     015476' 012746  010576'                                        MOV     #FORM61,-(SP)
     015502' 012746  000001                                         MOV     #1,-(SP)
     015506' 010600                                                 MOV     SP,R0
     015510' 104414                                                 TRAP    C$PNTB
     015512' 062706  000004                                         ADD     #4,SP
     015516'                                PRINTB  #FORM27,#52525,PCBB+4
     015516' 013746  000612'                                        MOV     PCBB+4,-(SP)
     015522' 012746  052525                                         MOV     #52525,-(SP)
     015526' 012746  006166'                                        MOV     #FORM27,-(SP)
     015532' 012746  000003                                         MOV     #3,-(SP)
     015536' 010600                                                 MOV     SP,R0
```

```
015540' 104414                                                      TRAP    C$PNTB
015542' 062706  000010                                              ADD     #10,SP
015546'                         PRINTB  #FORM57,PCBB
015546' 013746  000606'                                             MOV     PCBB,-(SP)
015552' 012746  010270'                                             MOV     #FORM57,-(SP)
015556' 012746  000002                                              MOV     #2,-(SP)
015562' 010600                                                      MOV     SP,R0
015564' 104414                                                      TRAP    C$PNTB
015566' 062706  000006                                              ADD     #6,SP
015572'                 ENDMSG
015572'                                                     L10045:
015572' 104423                                                      TRAP    C$MSG

015574'                 BGNMSG  MSG33
015574'                                                     MSG33::
015574'                         PRINTB  #FORM62,@BITNAM
015574' 017746  162510                                              MOV     @BITNAM,-(SP)
015600' 012746  010661'                                             MOV     #FORM62,-(SP)
015604' 012746  000002                                              MOV     #2,-(SP)
015610' 010600                                                      MOV     SP,R0
015612' 104414                                                      TRAP    C$PNTB
015614' 062706  000006                                              ADD     #6,SP
015620'                 ENDMSG
015620'                                                     L10046:
015620' 104423                                                      TRAP    C$MSG

015622'                 BGNMSG  MSG34
015622'                                                     MSG34::
015622'                         PRINTB  #FORM63,R4
015622' 010446                                                      MOV     R4,-(SP)
015624' 012746  010734'                                             MOV     #FORM63,-(SP)
015630' 012746  000002                                              MOV     #2,-(SP)
015634' 010600                                                      MOV     SP,R0
015636' 104414                                                      TRAP    C$PNTB
015640' 062706  000006                                              ADD     #6,SP
015644'                         PRINTB  #FORM28,STMSG
015644' 013746  024406'                                             MOV     STMSG,-(SP)
015650' 012746  006236'                                             MOV     #FORM28,-(SP)
015654' 012746  000002                                              MOV     #2,-(SP)
015660' 010600                                                      MOV     SP,R0
015662' 104414                                                      TRAP    C$PNTB
015664' 062706  000006                                              ADD     #6,SP
015670'                 ENDMSG
015670'                                                     L10047:
015670' 104423                                                      TRAP    C$MSG

015672'                 BGNMSG  MSG35
015672'                                                     MSG35::
015672'                         PRINTB  #FORM59,PCBB
015672' 013746  000606'                                             MOV     PCBB,-(SP)
015676' 012746  010416'                                             MOV     #FORM59,-(SP)
015702' 012746  000002                                              MOV     #2,-(SP)
015706' 010600                                                      MOV     SP,R0
015710' 104414                                                      TRAP    C$PNTB
015712' 062706  000006                                              ADD     #6,SP
015716'                         PRINTB  #FORM60
```

```
015716' 012746 010511'                                           MOV     #FORM60,-(SP)
015722' 012746 000001                                            MOV     #1,-(SP)
015726' 010600                                                   MOV     SP,R0
015730' 104414                                                   TRAP    C$PNTB
015732' 062706 000004                                            ADD     #4,SP
015736'                       PRINTB  #FORM78,PCBB,PCBB+2
015736' 013746 000610'                                           MOV     PCBB+2,-(SP)
015742' 013746 000606'                                           MOV     PCBB,-(SP)
015746' 012746 012132'                                           MOV     #FORM78,-(SP)
015752' 012746 000003                                            MOV     #3,-(SP)
015756' 010600                                                   MOV     SP,R0
015760' 104414                                                   TRAP    C$PNTB
015762' 062706 000010                                            ADD     #10,SP
015766'               ENDMSG
015766'                                                  L10050:
015766' 104423                                                   TRAP    C$MSG

015770'               BGNMSG  MSG36
015770'                                                  MSG36::
015770'                       PRINTB  #FORM66
015770' 012746 011072'                                           MOV     #FORM66,-(SP)
015774' 012746 000001                                            MOV     #1,-(SP)
016000' 010600                                                   MOV     SP,R0
016002' 104414                                                   TRAP    C$PNTB
016004' 062706 000004                                            ADD     #4,SP
016010'               ENDMSG
016010'                                                  L10051:
016010' 104423                                                   TRAP    C$MSG

016012'               BGNMSG  MSG37
016012'                                                  MSG37::
016012'                       PRINTB  #FORM67
016012' 012746 011157'                                           MOV     #FORM67,-(SP)
016016' 012746 000001                                            MOV     #1,-(SP)
016022' 010600                                                   MOV     SP,R0
016024' 104414                                                   TRAP    C$PNTB
016026' 062706 000004                                            ADD     #4,SP
016032'               ENDMSG
016032'                                                  L10052:
016032' 104423                                                   TRAP    C$MSG

016034'               BGNMSG  MSG38
016034'                                                  MSG38::
016034'                       PRINTB  #FORM68
016034' 012746 011243'                                           MOV     #FORM68,-(SP)
016040' 012746 000001                                            MOV     #1,-(SP)
016044' 010600                                                   MOV     SP,R0
016046' 104414                                                   TRAP    C$PNTB
016050' 062706 000004                                            ADD     #4,SP
016054'               ENDMSG
016054'                                                  L10053:
016054' 104423                                                   TRAP    C$MSG

016056'               BGNMSG  MSG39
016056'                                                  MSG39::
016056'                       PRINTB  #FORM69
```

```
016056' 012746  011340'                                          MOV     #FORM69,-(SP)
016062' 012746  000001                                           MOV     #1,-(SP)
016066' 010600                                                   MOV     SP,R0
016070' 104414                                                   TRAP    C$PNTB
016072' 062706  000004                                           ADD     #4,SP
016076'                   ENDMSG
016076'                                                 L10054:
016076' 104423                                                   TRAP    C$MSG

016100'                   BGNMSG  MSG40
016100'                                                 MSG40::
016100'                           PRINTB  #FORM56
016100' 012746  010215'                                          MOV     #FORM56,-(SP)
016104' 012746  000001                                           MOV     #1,-(SP)
016110' 010600                                                   MOV     SP,R0
016112' 104414                                                   TRAP    C$PNTB
016114' 062706  000004                                           ADD     #4,SP
016120'                           PRINTB  #FORM59,#100000
016120' 012746  100000                                           MOV     #100000,-(SP)
016124' 012746  010416'                                          MOV     #FORM59,-(SP)
016130' 012746  000002                                           MOV     #2,-(SP)
016134' 010600                                                   MOV     SP,R0
016136' 104414                                                   TRAP    C$PNTB
016140' 062706  000006                                           ADD     #6,SP
016144'                           PRINTB  #FORM60,#100000,PCBB
016144' 013746  000606'                                          MOV     PCBB,-(SP)
016150' 012746  100000                                           MOV     #100000,-(SP)
016154' 012746  010511'                                          MOV     #FORM60,-(SP)
016160' 012746  000003                                           MOV     #3,-(SP)
016164' 010600                                                   MOV     SP,R0
016166' 104414                                                   TRAP    C$PNTB
016170' 062706  000010                                           ADD     #10,SP
016174'                   ENDMSG
016174'                                                 L10055:
016174' 104423                                                   TRAP    C$MSG

016176'                   BGNMSG  MSG41
016176'                                                 MSG41::
016176'                           PRINTB  #FORM61
016176' 012746  010576'                                          MOV     #FORM61,-(SP)
016202' 012746  000001                                           MOV     #1,-(SP)
016206' 010600                                                   MOV     SP,R0
016210' 104414                                                   TRAP    C$PNTB
016212' 062706  000004                                           ADD     #4,SP
016216'                           PRINTB  #FORM27,#52525,PCBB+2
016216' 013746  000610'                                          MOV     PCBB+2,-(SP)
016222' 012746  052525                                           MOV     #52525,-(SP)
016226' 012746  006166'                                          MOV     #FORM27,-(SP)
016232' 012746  000003                                           MOV     #3,-(SP)
016236' 010600                                                   MOV     SP,R0
016240' 104414                                                   TRAP    C$PNTB
016242' 062706  000010                                           ADD     #10,SP
016246'                   ENDMSG
016246'                                                 L10056:
016246' 104423                                                   TRAP    C$MSG

```

```
016250'                          BGNMSG  MSG42
016250'                                                         MSG42::
016250'                                  PRINTB  #FORM70,PCBB+2
016250' 013746  000610'                                                    MOV     PCBB+2,-(SP)
016254' 012746  011456'                                                    MOV     #FORM70,-(SP)
016260' 012746  000002                                                     MOV     #2,-(SP)
016264' 010600                                                             MOV     SP,R0
016266' 104414                                                             TRAP    C$PNTB
016270' 062706  000006                                                     ADD     #6,SP
016274'                                  PRINTB  #FORM71,PCBB
016274' 013746  000606'                                                    MOV     PCBB,-(SP)
016300' 012746  011520'                                                    MOV     #FORM71,-(SP)
016304' 012746  000002                                                     MOV     #2,-(SP)
016310' 010600                                                             MOV     SP,R0
016312' 104414                                                             TRAP    C$PNTB
016314' 062706  000006                                                     ADD     #6,SP
016320'                                  PRINTB  #FORM33,R1
016320' 010146                                                             MOV     R1,-(SP)
016322' 012746  006441'                                                    MOV     #FORM33,-(SP)
016326' 012746  000002                                                     MOV     #2,-(SP)
016332' 010600                                                             MOV     SP,R0
016334' 104414                                                             TRAP    C$PNTB
016336' 062706  000006                                                     ADD     #6,SP
016342'                                  PRINTB  #FORM36,PCBB+4
016342' 013746  000612'                                                    MOV     PCBB+4,-(SP)
016346' 012746  006662'                                                    MOV     #FORM36,-(SP)
016352' 012746  000002                                                     MOV     #2,-(SP)
016356' 010600                                                             MOV     SP,R0
016360' 104414                                                             TRAP    C$PNTB
016362' 062706  000006                                                     ADD     #6,SP
016366'                                  PRINTB  #FORM27,PCBB+6,PCBB+10
016366' 013746  000616'                                                    MOV     PCBB+10,-(SP)
016372' 013746  000614'                                                    MOV     PCBB+6,-(SP)
016376' 012746  006166'                                                    MOV     #FORM27,-(SP)
016402' 012746  000003                                                     MOV     #3,-(SP)
016406' 010600                                                             MOV     SP,R0
016410' 104414                                                             TRAP    C$PNTB
016412' 062706  000010                                                     ADD     #10,SP
016416'                          ENDMSG
016416'                                                         L10057:
016416' 104423                                                             TRAP    C$MSG

016420'                          BGNMSG  MSG43
016420'                                                         MSG43::
016420'                                  PRINTB  #FORM72
016420' 012746  011562'                                                    MOV     #FORM72,-(SP)
016424' 012746  000001                                                     MOV     #1,-(SP)
016430' 010600                                                             MOV     SP,R0
016432' 104414                                                             TRAP    C$PNTB
016434' 062706  000004                                                     ADD     #4,SP
016440'                          ENDMSG
016440'                                                         L10060:
016440' 104423                                                             TRAP    C$MSG

016442'                          BGNMSG  MSG44
016442'                                                         MSG44::
```

```
016442' 032737  100000  000670'        BIT     #NPRERR,ERRINT          ;NPR ERROR OCCUR?
016450' 001410                         BEQ     10$                     ;NO
016452'                                PRINTB  #FORM73                 ;YES, PRINT NPR ERROR MESSAGE
016452' 012746  011634'                                                                MOV     #FORM73,-(SP)
016456' 012746  000001                                                                 MOV     #1,-(SP)
016462' 010600                                                                         MOV     SP,R0
016464' 104414                                                                         TRAP    C$PNTB
016466' 062706  000004                                                                 ADD     #4,SP
016472' 032737  040000  000670' 10$:   BIT     #NXMERR,ERRINT          ;NON-EXISTANT MEMORY OCCUR?
016500' 001410                         BEQ     20$                     ;NO
016502'                                PRINTB  #FORM74                 ;YES, PRINT NON-EXISTANT MEMORY MESSAGE
016502' 012746  011677'                                                                MOV     #FORM74,-(SP)
016506' 012746  000001                                                                 MOV     #1,-(SP)
016512' 010600                                                                         MOV     SP,R0
016514' 104414                                                                         TRAP    C$PNTB
016516' 062706  000004                                                                 ADD     #4,SP
016522' 032737  020000  000670' 20$:   BIT     #UNIERR,ERRINT          ;UNEXPECTED INTERRUPT OCCUR?
016530' 001410                         BEQ     30$                     ;NO
016532'                                PRINTB  #FORM75                 ;YES, PRINT UNEXPECTED INTERRUPT MESSAGE
016532' 012746  011754'                                                                MOV     #FORM75,-(SP)
016536' 012746  000001                                                                 MOV     #1,-(SP)
016542' 010600                                                                         MOV     SP,R0
016544' 104414                                                                         TRAP    C$PNTB
016546' 062706  000004                                                                 ADD     #4,SP
016552' 032737  010000  000670' 30$:   BIT     #PARERR,ERRINT          ;PARITY ERROR OCCUR?
016560' 001410                         BEQ     40$                     ;NO
016562'                                PRINTB  #FORM72                 ;YES, PRINT PARITY ERROR MESSAGE
016562' 012746  011562'                                                                MOV     #FORM72,-(SP)
016566' 012746  000001                                                                 MOV     #1,-(SP)
016572' 010600                                                                         MOV     SP,R0
016574' 104414                                                                         TRAP    C$PNTB
016576' 062706  000004                                                                 ADD     #4,SP
016602'                         40$:
016602'                         ENDMSG
016602'                                                                        L10061:
016602' 104423                                                                         TRAP    C$MSG

016604'                         BGNMSG  MSG45
016604'                                                                        MSG45::
016604'                                PRINTB  #FORM79
016604' 012746  012172'                                                                MOV     #FORM79,-(SP)
016610' 012746  000001                                                                 MOV     #1,-(SP)
016614' 010600                                                                         MOV     SP,R0
016616' 104414                                                                         TRAP    C$PNTB
016620' 062706  000004                                                                 ADD     #4,SP
016624'                                PRINTB  #FORM80
016624' 012746  012241'                                                                MOV     #FORM80,-(SP)
016630' 012746  000001                                                                 MOV     #1,-(SP)
016634' 010600                                                                         MOV     SP,R0
016636' 104414                                                                         TRAP    C$PNTB
016640' 062706  000004                                                                 ADD     #4,SP
016644'                                PRINTB  #FORM81
016644' 012746  012320'                                                                MOV     #FORM81,-(SP)
016650' 012746  000001                                                                 MOV     #1,-(SP)
016654' 010600                                                                         MOV     SP,R0
016656' 104414                                                                         TRAP    C$PNTB
```

```
016660' 062706  000004                                                  ADD     #4,SP
016664'                         ENDMSG
016664'                                                         L10062:
016664' 104423                                                          TRAP    C$MSG

016666'                         BGNMSG  MSG46
016666'                                                         MSG46::
016666'                                 PRINTB  #FORM82
016666' 012746  012363'                                                 MOV     #FORM82,-(SP)
016672' 012746  000001                                                  MOV     #1,-(SP)
016676' 010600                                                          MOV     SP,R0
016700' 104414                                                          TRAP    C$PNTB
016702' 062706  000004                                                  ADD     #4,SP
016706'                         ENDMSG
016706'                                                         L10063:
016706' 104423                                                          TRAP    C$MSG
```

```
                                .SBTTL  GLOBAL SUBROUTINES SECTION

                                ;++
                                ; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
                                ; THAT ARE USED IN MORE THAN ONE TEST.
                                ;--
                                .SBTTL  32 BIT CRC CALCULATOR

                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;     SUBROUTINE TO CALCULATE A 32 BIT CRC ON A BLOCK OF DATA
                                ;
                                ; INPUTS:
                                ;       R0 = ADDRESS OF DATA BLOCK
                                ;       R2 = BYTE COUNT
                                ;
                                ; IMPLICIT INPUTS: NONE
                                ;
                                ; OUTPUTS:
                                ;       R4 = CRC HIGH WORD
                                ;       R5 = CRC LOW WORD
                                ;
                                ; SUBORDINATE ROUTINES USED: GETCRC
                                ;
                                ; FUNCTIONAL SIDE EFFECTS: NONE
                                ;
                                ; CALLING SEQUENCE: PUT ADDRESS OF DATA TO PERFORM CRC ON IN R0
                                ;                   PUT NUMBER OF BYTES TO CHECK IN R2
                                ;                   JSR     PC,BLKCRC
                                ;
                                ;--

016710'                         CRC32::
016710' 010046                          MOV     R0,-(SP)        ;SAVE REGISTERS 0-3
016712' 010146                          MOV     R1,-(SP)
016714' 010246                          MOV     R2,-(SP)
016716' 010346                          MOV     R3,-(SP)
016720' 012704  177777                  MOV     #INITH,R4       ;INITIAL CRC HIGH WORD
016724' 012705  177777                  MOV     #INITL,R5       ;INITIAL CRC LOW WORD
016730' 112001                  10$:    MOVB    (R0)+,R1        ;GET NEXT BYTE OF DATA
016732' 004737  016752'                 JSR     PC,GETCRC       ;CALCULATE THE CRC
016736' 077204                          SOB     R2,10$          ;LOOP TILL DONE
016740' 012603                          MOV     (SP)+,R3        ;RESTORE REGISTERS
016742' 012602                          MOV     (SP)+,R2
016744' 012601                          MOV     (SP)+,R1
016746' 012600                          MOV     (SP)+,R0
016750' 000207                          RTS     PC              ;RETURN
```

```
                                ;++
                                ;BYTE WISE 32-BIT CRC CALCULATOR
                                ;
                                ;INPUTS:
                                ;       R1 = NEW BYTE TO ADD TO CRC
                                ;       R4,R5 = CURRENT PARTIAL CRC CODE
                                ;
                                ;OUTPUTS:
                                ;       R4,R5 = UPDATED CRC
                                ;
                                ;NOTE: THIS ROUTINE IS ONLY USED BY BLKCRC
                                ;
                                ;--

 016752'                        GETCRC:
 016752' 010146                         MOV     R1,-(SP)        ;SAVE R1-3
 016754' 010246                         MOV     R2,-(SP)
 016756' 010346                         MOV     R3,-(SP)
 016760' 042701  177400                 BIC     #^C377,R1       ;CLEAR HIGH BYTE
 016764' 074105                         XOR     R1,R5           ;MERGE NEW BYTE WITH OLD CRC
 016766' 012702  166670                 MOV     #POLYH,R2       ;GET CRC POLYNOMIAL HIGH WORD
 016772' 012703  101440                 MOV     #POLYL,R3       ;GET CRC POLYNOMIAL LOW WORD
 016776' 012701  000010                 MOV     #8.,R1          ;LOOP COUNT
 017002' 000241                 1$:     CLC                     ;CLEAR THE CARRY
 017004' 006004                         ROR     R4              ;SHIFT RIGHT THE CRC
 017006' 006005                         ROR     R5              ;32 BITS WORTH
 017010' 103002                         BCC     2$              ;SKIP IF BIT 0 NOT SET
 017012' 074204                         XOR     R2,R4           ;EXCLUSIVE OR IN THE POLY
 017014' 074305                         XOR     R3,R5           ;BOTH HIGH AND LOW WORDS
 017016' 077107                 2$:     SOB     R1,1$           ;AND LOOP ON ALL 8 BITS
 017020' 012603                         MOV     (SP)+,R3        ;RESTORE REGISTERS
 017022' 012602                         MOV     (SP)+,R2
 017024' 012601                         MOV     (SP)+,R1
 017026' 000207                         RTS     PC              ;RETURN
```

```

                                         .SBTTL  16 BIT CRC CALCULATOR

                                         ;*****************************************************************
                                         ;
                                         ;       SUBROUTINE - CRC16
                                         ;
                                         ;       THIS SUBROUTINE CALCULATES A 16 BIT CRC
                                         ;       ON A BLOCK OF DATA.
                                         ;
                                         ;       INPUTS:         R0 = ADDRESS OF DATA BLOCK
                                         ;                       R2 = BYTE COUNT
                                         ;                       R4 = INITIAL CRC VALUE
                                         ;
                                         ;       OUTPUTS:        R4 = CRC
                                         ;
                                         ;       CALLING SEQUENCE:
                                         ;               JSR     PC,CRC16
                                         ;
                                         ;*****************************************************************

 017030'                                 CRC16::
 017030' 010046                                  MOV     R0,-(SP)                ; SAVE R0
 017032' 010146                                  MOV     R1,-(SP)                ; SAVE R1
 017034' 010246                                  MOV     R2,-(SP)                ; SAVE R2
 017036' 010346                                  MOV     R3,-(SP)                ; SAVE R3
 017040' 010546                                  MOV     R5,-(SP)                ; SAVE R5
 017042' 010203                                  MOV     R2,R3                   ; R3 = BYTE COUNT
 017044' 010002                                  MOV     R0,R2                   ; R2 = ADDRESS OF DATA BLOCK
 017046' 012705  120001                          MOV     #POLY16,R5              ; CRC POLYNOMIAL
 017052' 112200                          1$:     MOVB    (R2)+,R0                ; GET NEXT BYTE
 017054' 042700  177400                          BIC     #^C377,R0               ; CLEAR HIGH BYTE
 017060' 074004                                  XOR     R0,R4                   ; MERGE BYTE WITH OLD_CRC
 017062' 012701  000010                          MOV     #8.,R1                  ; LOOP COUNT
 017066' 000241                          2$:     CLC                             ; CLEAR CARRY
 017070' 006004                                  ROR     R4                      ; SHIFT RIGHT THE CRC
 017072' 103001                                  BCC     3$                      ; SKIP IF BIT ZERO NOT SET
 017074' 074504                                  XOR     R5,R4                   ; EXCLUSIVE OR IN THE POLY
 017076' 077105                          3$:     SOB     R1,2$                   ; AND LOOP ON ALL 8 BITS
 017100' 077314                                  SOB     R3,1$
 017102' 012605                                  MOV     (SP)+,R5                ; RESTORE R5
 017104' 012603                                  MOV     (SP)+,R3                ; RESTORE R3
 017106' 012602                                  MOV     (SP)+,R2                ; RESTORE R2
 017110' 012601                                  MOV     (SP)+,R1                ; RESTORE R1
 017112' 012600                                  MOV     (SP)+,R0                ; RESTORE R0
 017114' 000207                                  RTS     PC                      ; AND RETURN

```

```
                                        .SBTTL  HEXIDECIMAL CONVERTER FOR DEFAULT PHYSICAL ADDRESS

                                        ;*************************************************************************
                                        ;
                                        ;       SUBROUTINE - HEXDPA
                                        ;
                                        ;       THIS SUBROUTINE LOADS DEFADR WITH THE ASCII HEX VALUE
                                        ;       FOR THE DEFAULT PHYSICAL ADDRESS DPA.
                                        ;
                                        ;       INPUTS:         NONE
                                        ;
                                        ;       IMPLICIT
                                        ;       INPUTS:         DPA = DEFAULT PHYSICAL ADDRESS
                                        ;
                                        ;       OUTPUTS:        DEFADR = ASCII HEX VALUE FOR DPA
                                        ;
                                        ;       CALLING SEQUENCE:
                                        ;               JSR     PC,HEXDPA
                                        ;
                                        ;*************************************************************************

 017116'                                HEXDPA::
 017116' 010046                                 MOV     R0,-(SP)                ; SAVE R0
 017120' 010346                                 MOV     R3,-(SP)                ; SAVE R3
 017122' 010546                                 MOV     R5,-(SP)                ; SAVE R5
                                        ;
 017124' 012700  000006                         MOV     #6,R0                   ; DO LOOP = 6 BYTES
 017130' 012703  052625'                        MOV     #DEFADR,R3              ; POINT TO ASCII MESSAGE
 017134' 012705  052542'                        MOV     #DPA,R5                 ; POINT TO DEFAULT PHYSICAL ADDR
                                        ;
 017140' 112537  052560'                10$:    MOVB    (R5)+,HEXDAT            ; LOAD BYTE FOR CONVERSION
 017144' 004737  017200'                        JSR     PC,HEXH                 ; CONVERT HIGH NIBBLE
 017150' 113723  052561'                        MOVB    HEXVAL,(R3)+            ; LOAD INTO ASCII MESSAGE
 017154' 004737  017236'                        JSR     PC,HEXL                 ; CONVERT LOW NIBBLE
 017160' 113723  052561'                        MOVB    HEXVAL,(R3)+            ; LOAD INTO ASCII MESSAGE
 017164' 105723                                 TSTB    (R3)+                   ; SKIP OVER HYPHEN IN MESSAGE
 017166' 077014                                 SOB     R0,10$                  ; LOOP TILL ALL 6 BYTES ARE DONE
                                        ;
 017170' 012605                                 MOV     (SP)+,R5                ; RESTORE R5
 017172' 012603                                 MOV     (SP)+,R3                ; RESTORE R3
 017174' 012600                                 MOV     (SP)+,R0                ; RESTORE R0
 017176' 000207                                 RTS     PC                      ; AND RETURN
```

```

                                ;***********************************************************************
                                ;
                                ;               SUBROUTINE - HEXH
                                ;
                                ;               THIS SUBROUTINE LOADS HEXVAL WITH THE ASCII HEX VALUE
                                ;               FOR THE HIGH NIBBLE IN HEXDAT
                                ;
                                ;               INPUTS:         NONE
                                ;
                                ;               IMPLICIT
                                ;               INPUTS:         HEXDAT = BYTE TO BE CONVERTED
                                ;
                                ;               OUTPUTS:        HEXVAL = ASCII HEX VALUE FOR THE HIGH NIBBLE
                                ;
                                ;               CALLING SEQUENCE:
                                ;                       JSR     PC,HEXH
                                ;
                                ;***********************************************************************

 017200'                        HEXH::
 017200' 010146                         MOV     R1,-(SP)                ; SAVE R1
                                ;
 017202' 013701  052560'                MOV     HEXDAT,R1               ; LOAD DATA FOR CONVERSION
 017206' 042701  177417                 BIC     #177417,R1              ; MASK HIGH NIBBLE
                                ;
 017212' 006201                         ASR     R1                      ; SHIFT RIGHT
 017214' 006201                         ASR     R1
 017216' 006201                         ASR     R1
 017220' 006201                         ASR     R1
                                ;
 017222' 062701  052651'                ADD     #HEXTBL,R1              ; GET INDEX INTO HEXTBL
 017226' 111137  052561'                MOVB    (R1),HEXVAL             ; AND LOAD HEXVAL
                                ;
 017232' 012601                         MOV     (SP)+,R1                ; RESTORE R1
 017234' 000207                         RTS     PC                      ; AND RETURN
```

```
                                ;***********************************************************************
                                ;
                                ;       SUBROUTINE - HEXL
                                ;
                                ;       THIS SUBROUTINE LOADS HEXVAL WITH THE ASCII HEX VALUE
                                ;       FOR THE LOW NIBBLE IN HEXDAT
                                ;
                                ;       INPUTS:         NONE
                                ;
                                ;       IMPLICIT
                                ;       INPUTS:         HEXDAT = BYTE TO BE CONVERTED
                                ;
                                ;       OUTPUTS:        HEXVAL = ASCII HEX VALUE FOR THE LOW NIBBLE
                                ;
                                ;       CALLING SEQUENCE:
                                ;               JSR     PC,HEXL
                                ;
                                ;***********************************************************************

 017236'                        HEXL::
 017236' 010146                         MOV     R1,-(SP)                ; SAVE R1
                                ;
 017240' 013701  052560'                MOV     HEXDAT,R1               ; LOAD DATA FOR CONVERSION
 017244' 042701  177760                 BIC     #177760,R1              ; MASK LOW NIBBLE
                                ;
 017250' 062701  052651'                ADD     #HEXTBL,R1              ; GET INDEX INTO HEXTBL
 017254' 111137  052561'                MOVB    (R1),HEXVAL             ; AND LOAD HEXVAL
                                ;
 017260' 012601                         MOV     (SP)+,R1                ; RESTORE R1
 017262' 000207                         RTS     PC                      ; AND RETURN
```

```
                                        .SBTTL  TURN ON THE CLOCK

                                        ;*************************************************************************
                                        ;
                                        ;THIS ROUTINE TURNS ON THE CLOCK
                                        ;
                                        ;*************************************************************************

017264'                                 TIMON:: SETPRI  #PRIO5                  ;SET PROCESSOR PRIORITY TO 5
017264' 012700  000240                                                                  MOV     #PRIO5,RO
017270' 104441                                                                          TRAP    C$SPRI
017272' 012777  000100  160774                  MOV     #IE,@CLKCSR             ;ENABLE CLOCK INTERRUPTS
017300' 000207                                  RTS     PC


                                        ;*************************************************************************
                                        ;
                                        ;THIS ROUTINE TURNS THE CLOCK OFF
                                        ;
                                        ;*************************************************************************

017302' 005077  160766                  TIMOFF::CLR     @CLKCSR                 ;CLEAR INTERRUPT ENABLE
017306'                                         SETPRI  #PRIO7                  ;PUT PRIORITY BACK UP
017306' 012700  000340                                                                  MOV     #PRIO7,RO
017312' 104441                                                                          TRAP    C$SPRI
017314' 000207                                  RTS     PC
```

```
                                        .SBTTL  CHECK FOR DONE INTERRUPT

                                        ;*************************************************************************
                                        ;
                                        ;FUNCTIONAL DESCRIPTION:
                                        ;       ROUTINE TO WAIT FOR THE 'DONE INTERRUPT' BIT TO SET IN PCSRO
                                        ;
                                        ;INPUTS: NONE
                                        ;
                                        ;IMPLICIT INPUTS: METER
                                        ;
                                        ;OUTPUTS: C BIT SET IN PSW IF 'DNI' NOT SET
                                        ;         C BIT CLEAR IN PSW IF 'DNI' SET
                                        ;
                                        ;SUBORDINATE ROUTINES USED: TIMON, TIMOFF
                                        ;
                                        ;FUNCTIONAL SIDE EFFECTS: PSW CHANGED, LINE CLOCK INTERRUPTS ARE ENABLED
                                        ;
                                        ;CALLING SEQUENCE: PUT NUMBER OF CLOCK TICKS TO WAIT FOR IN--->METER
                                        ;                  JSR PC,CHKDNI
                                        ;
                                        ;*************************************************************************

017316' 004737  017264'                 CHKDNI::JSR     PC,TIMON                ;TURN ON THE LINE CLOCK
017322' 032777  004000  161006          10$:    BIT     #DNI,@PCSRO             ;IS 'DNI' SET?
017330' 001010                                  BNE     20$                     ;YES
017332'                                         BREAK                           ;RETURN TO THE DRS FOR A MOMENT
017332' 104422                                                                          TRAP    C$BRK
017334' 005737  000332'                         TST     METER                   ;HAS THE TIME EXPIRED?
017340' 001370                                  BNE     10$                     ;NO, KEEP WAITING
017342' 004737  017302'                         JSR     PC,TIMOFF               ;TURN THE CLOCK OFF
017346' 000261                                  SEC                             ;INDICATE 'DNI' DID NOT SET
017350' 000403                                  BR      30$                     ;LEAVE
017352' 004737  017302'                 20$:    JSR     PC,TIMOFF               ;STOP THE CLOCK
017356' 000241                                  CLC                             ;INDICATE 'DNI' SET
017360' 000207                          30$:    RTS     PC
```

```
                                        .SBTTL  CLEAR DONE INTERRUPT

                                        ;***************************************************************************
                                        ;
                                        ;FUNCTIONAL DESCRIPTION:
                                        ;       ROUTINE TO CLEAR THE 'DNI' BIT IN PCSRO
                                        ;
                                        ;INPUTS: NONE
                                        ;
                                        ;IMPLICIT INPUTS: PCSRO
                                        ;
                                        ;OUTPUTS: C BIT SET IN PCSW IF 'DNI' WILL NOT CLEAR
                                        ;         C BIT CLEAR IN PSW IF 'DNI' CLEARED SUCCESFULLY
                                        ;
                                        ;SUBORDINATE ROUTINES CALLED: NONE
                                        ;
                                        ;FUNCTIONAL SIDE EFFECTS: PSW CHANGED
                                        ;CALLING SEQUENCE: JSR PC,CLRDNI
                                        ;
                                        ;*********************************************************************
   017362' 112777  000010  160756      CLRDNI::MOVB    #DNIB,@PCSROUB          ;CLEAR 'DNI' BIT            ;RSJ001
   017370' 032777  004000  160740              BIT     #DNI,@PCSRO             ;DID IS CLEAR?
   017376' 001402                              BEQ     10$                     ;YES
   017400' 000261                              SEC                             ;NO, INDICATE ERROR
   017402' 000401                              BR      20$
   017404' 000241                      10$:    CLC                             ;YES, INDICATE SUCCESS
   017406' 000207                      20$:    RTS     PC
```

```
                                        .SBTTL  CLEAR OUTSTANDING INTERRUPT BITS

                                        ;**********************************************************************8
                                        ;
                                        ;FUNCTIONAL DESCRIPTION:
                                        ;       ROUTINE TO CLEAR ALL INTERRUPT BITS IN PCSR0
                                        ;
                                        ;INPUTS: NONE
                                        ;
                                        ;IMPLICIT INPUTS: NONE
                                        ;
                                        ;OUTPUTS: C BIT SET IF UNABLE TO CLEAR A INTERRUPT BIT
                                        ;         C BIT CLEARED IF SUCCESSFUL
                                        ;
                                        ;SUBORDINATE ROUTINES USED: NONE
                                        ;
                                        ;FUNCTIONAL SIDE EFFECTS: ANY OUTSTANDING INTERRUPT IS CLEARED
                                        ;
                                        ;CALLING SEQUENCE: JSR PC,CLRINT
                                        ;
                                        ;*********************************************************************

017410' 032777  100000  160720  CLRINT: BIT     #SERI,@PCSR0            ;IS 'SERI' BIT SET?
017416' 001413                          BEQ     10$                     ;NO
017420' 112777  000200  160720          MOVB    #SERIB,@PCSR0UB         ;WRITE ONE TO CLEAR 'SERI'      ;RSJ001
017426' 032777  100000  160702          BIT     #SERI,@PCSR0            ;DID IT CLEAR?
017434' 001404                          BEQ     10$                     ;YES
017436' 012737  000736' 000310'         MOV     #$SERI,BITNAM           ;NO GET POINTER TO BIT NAME STRING
017444' 000531                          BR      70$                     ;LEAVE
017446' 032777  040000  160662  10$:    BIT     #PCEI,@PCSR0            ;IS 'PCEI' BIT SET?
017454' 001413                          BEQ     20$                     ;NO
017456' 112777  000100  160662          MOVB    #PCEIB,@PCSR0UB         ;YES, WRITE ONE TO CLEAR 'PCEI' ;RSJ001
017464' 032777  040000  160644          BIT     #PCEI,@PCSR0            ;DID IT CLEAR?
017472' 001404                          BEQ     20$                     ;YES
017474' 012737  000747' 000310'         MOV     #$PCEI,BITNAM           ;NO, GET POINTER TO BIT NAME STRING
017502' 000512                          BR      70$                     ;LEAVE
017504' 032777  020000  160624  20$:    BIT     #RXI,@PCSR0             ;IS 'RXI' BIT SET?
017512' 001413                          BEQ     30$                     ;NO
017514' 112777  000040  160624          MOVB    #RXIB,@PCSR0UB          ;YES, WRITE ONE TO CLEAR 'RXI'  ;RSJ001
017522' 032777  020000  160606          BIT     #RXI,@PCSR0             ;DID IT CLEAR?
017530' 001404                          BEQ     30$                     ;YES
017532' 012737  000760' 000310'         MOV     #$RXI,BITNAM            ;NO, GET POINTER TO BIT NAME STRING
017540' 000473                          BR      70$                     ;LEAVE
017542' 032777  010000  160566  30$:    BIT     #TXI,@PCSR0             ;IS 'TXI' BIT SET?
017550' 001413                          BEQ     40$                     ;NO
017552' 112777  000020  160566          MOVB    #TXIB,@PCSR0UB          ;YES, WRITE ONE TO CLEAR 'TXI'  ;RSJ001
017560' 032777  010000  160550          BIT     #TXI,@PCSR0             ;DID IT CLEAR?
017566' 001404                          BEQ     40$                     ;YES
017570' 012737  000770' 000310'         MOV     #$TXI,BITNAM            ;NO, GET POINTER TO BIT NAME STRING
017576' 000454                          BR      70$                     ;LEAVE
017600' 032777  004000  160530  40$:    BIT     #DNI,@PCSR0             ;IS 'DNI' BIT SET?
017606' 001413                          BEQ     50$                     ;NO
017610' 112777  000010  160530          MOVB    #DNIB,@PCSR0UB          ;YES, WRITE ONE TO CLEAR 'DNI'  ;RSJ001
017616' 032777  004000  160512          BIT     #DNI,@PCSR0             ;DID IT CLEAR?
017624' 001404                          BEQ     50$                     ;YES
017626' 012737  001000' 000310'         MOV     #$DNI,BITNAM            ;NO, GET POINTER TO BIT NAME STRING
```

```
017634' 000435                              BR      70$                 ;LEAVE
017636' 032777  002000  160472  50$:        BIT     #RCEI,@PCSRO        ;IS 'RCEI' BIT SET?
017644' 001413                              BEQ     60$                 ;NO
017646' 112777  000004  160472              MOVB    #RCEIB,@PCSROUB     ;WRITE ONE TO CLEAR 'RCEI'      ;RSJ001
017654' 032777  002000  160454              BIT     #RCEI,@PCSRO        ;DID IT CLEAR?
017662' 001404                              BEQ     60$                 ;YES
017664' 012737  001010' 000310'             MOV     #$RCEI,BITNAM       ;NO, GET POINTER TO BIT NAME STRING
017672' 000416                              BR      70$                 ;LEAVE
017674' 032777  000400  160434  60$:        BIT     #FATI,@PCSRO        ;IS 'FATI' BIT SET?
017702' 001426                              BEQ     80$                 ;NO
017704' 112777  000001  160434              MOVB    #FATIB,@PCSROUB     ;WRITE ONE TO CLEAR 'FATI'      ;RSJ001
017712' 032777  000400  160416              BIT     #FATI,@PCSRO        ;DID IS CLEAR?
017720' 001417                              BEQ     80$                 ;YES
017722' 012737  001021' 000310'             MOV     #$FATI,BITNAM       ;NO, GET POINTER TO BIT NAME STRING
017730'                         70$:        PRINTF  #FORM62,BITNAM      ;PRINT ERROR MESSAGE
017730' 013746  000310'                                                              MOV     BITNAM,-(SP)
017734' 012746  010661'                                                              MOV     #FORM62,-(SP)
017740' 012746  000002                                                               MOV     #2,-(SP)
017744' 010600                                                                       MOV     SP,R0
017746' 104417                                                                       TRAP    C$PNTF
017750' 062706  000006                                                               ADD     #6,SP
017754' 000261                              SEC                         ;INDICATE ERROR TO CALLER
017756' 000401                              BR      100$                ;LEAVE
017760' 000241                  80$:        CLC                         ;INDICATE SUCCESS
017762' 000207                  100$:       RTS     PC
```

```
4396                                .SBTTL  PRINT PCSR'S                                                  ;MAC001
4397                                                                                                      ;MAC001
4398                                ;*********************************************************************;MAC001
4399                                ;                                                                     ;MAC001
4400                                ;FUNCTIONAL DESCRIPTION:                                              ;MAC001
4401                                ;                                                                     ;MAC001
4402                                ;       ROUTINE TO PRINT THE CONTENTS OF THE PCSR'S                   ;MAC001
4403                                ;                                                                     ;MAC001
4404                                ;INPUTS: NONE                                                         ;MAC001
4405                                ;                                                                     ;MAC001
4406                                ;OUTPUTS: CONTENTS OF PCSR'S 0,1,2,3 ARE PLACED IN LOCATIONS PCSR0C,   ;MAC001
4407                                ;               PCSR1C,PCSR2C,PCSR3C RESPECTIVELY AND PRINTED         ;MAC001
4408                                ;                                                                     ;MAC001
4409                                ;CALLING SEQUENCE: JSR PC,PRNPCR                                      ;MAC001
4410                                ;                                                                     ;MAC001
4411                                ;*********************************************************************;MAC001
4412                                                                                                      ;MAC001
4413 017764' 010146                 PRNPCR: MOV     R1,-(SP)                ;SAVE R1 AND R2               ;MAC001
4414 017766' 010246                         MOV     R2,-(SP)                                              ;MAC001
4415 017770' 012701 000360'                 MOV     #PCSR3C+2,R1            ;POINT TO BOTTOM OF STORAGE   ;MAC001
4416 017774' 017741 160344                  MOV     @PCSR3,-(R1)            ;SAVE PCSR3 IN TABLE          ;MAC001
4417 020000' 017741 160336                  MOV     @PCSR2,-(R1)            ;SAVE PCSR2 IN TABLE          ;MAC001
4418 020004' 017741 160330                  MOV     @PCSR1,-(R1)            ;SAVE PCSR1 IN TABLE          ;MAC001
4419 020010' 017741 160322                  MOV     @PCSR0,-(R1)            ;SAVE PCSR0 IN TABLE          ;MAC001
4420 020014' 005002                         CLR     R2                      ;WHICH PCSR TO PRINT          ;MAC001
4421 020016'                        10$:    PRINTX  #FORM83,R2,(R1)+        ;PRINT CONTENTS OF A PCSR     ;MAC001
4422 020016' 012146                                                                 MOV     (R1)+,-(SP)
4423 020020' 010246                                                                 MOV     R2,-(SP)
4424 020022' 012746 012453'                                                         MOV     #FORM83,-(SP)
4425 020026' 012746 000003                                                          MOV     #3,-(SP)
4426 020032' 010600                                                                 MOV     SP,R0
4427 020034' 104415                                                                 TRAP    C$PNTX
4428 020036' 062706 000010                                                          ADD     #10,SP
4429 020042' 005202                         INC     R2                      ;NEXT PCSR                    ;MAC001
4430 020044' 020227 000004                  CMP     R2,#4                   ;DONE LAST PCSR?              ;MAC001
4431 020050' 001362                         BNE     10$                     ;NOT YET                      ;MAC001
4432 020052' 012602                         MOV     (SP)+,R2                ;RESTORE R2                   ;MAC001
4433 020054' 012601                         MOV     (SP)+,R1                ;RESTORE R1                   ;MAC001
4434 020056' 000207                         RTS     PC                      ;RETURN                       ;MAC001
```

```
                                        .SBTTL  CHECK MICROMONITOR

                                        ;*******************************************************************************
                                        ;
                                        ;FUNCTIONAL DESCRIPTION:
                                        ;       ROUTINE TO WAIT FOR THE MICROCODE TO ENTER THE MICROMONITOR
                                        ;
                                        ;INPUTS: NONE
                                        ;
                                        ;IMPLICIT INPUTS: PCSR1
                                        ;
                                        ;OUTPUTS: C BIT SET IN PSW IF TIMEOUT WAITING FOR MICROMONITOR
                                        ;         C BIT CLEAR IN PSW IF MICROCODE IN MICROMONITOR
                                        ;
                                        ;SUBORDINATE ROUTINES CALLED: TIMON, TIMOFF
                                        ;
                                        ;FUNCTIONAL SIDE EFFECTS: PSW CHANGED, LINE CLOCK INTERRUPTS ARE ENABLED
                                        ;
                                        ;CALLING SEQUENCE: JSR PC,CHKMON
                                        ;
                                        ;*******************************************************************************

020060' 012737  000077  000332' CHKMON::MOV     #1*SECOND,METER         ;TIMEOUT PERIOD IS 1 SECOND
020066' 004737  017264'                 JSR     PC,TIMON                ;TURN ON THE CLOCK
020072' 122777  000001  160240  10$:    CMPB    #INMON,@PCSR1           ;IS THE MICROCODE IN THE MICROMONITOR?
020100' 001410                          BEQ     20$                     ;YES
020102'                                 BREAK                           ;RETURN TO DRS FOR A MOMENT
020102' 104422                                                                          TRAP    C$BRK
020104' 005737  000332'                 TST     METER                   ;HAS THE TIMER EXPIRED?
020110' 001370                          BNE     10$                     ;NOT YET, KEEP CHECKING MICROCODE
020112' 004737  017302'                 JSR     PC,TIMOFF               ;TIMER HAS EXPIRED TURN OFF THE TIMER
020116' 000261                          SEC                             ;INDICATE ERROR TO CALLER
020120' 000403                          BR      30$                     ;LEAVE
020122' 004737  017302'         20$:    JSR     PC,TIMOFF               ;STOP THE CLOCK
020126' 000241                          CLC                             ;INDICATE TO CALLER MICROCODE IS IN
                                                                        ;MICROMONITOR
020130' 000207                  30$:    RTS     PC
```

```
                                            .SBTTL  CHECK INTERRUPT ERROR BITS

                                            ;**********************************************************************
                                            ;
                                            ;FUNCTIONAL DESCRIPTION:
                                            ;       ROUTINE TO CHECK FOR ANY T11 INTERRUPT ERROR BITS IN PCSR0
                                            ;
                                            ;INPUTS: NONE
                                            ;
                                            ;IMPLICIT INPUTS: PCSR0
                                            ;
                                            ;OUTPUTS: C BIT SET IN PSW IF AN INTERRUPT BIT IS SET
                                            ;         C BIT CLEAR IN NO INTERRUPT BITS ARE SET
                                            ;         'ERRINT' CONTAINS COPY OF PCSR0
                                            ;
                                            ;SUBORDINATE ROUTINES CALLED:  NONE
                                            ;
                                            ;FUNCTIONAL SIDE EFFECTS: ANY INTERRUPT BIT SET IS CLEARED
                                            ;
                                            ;CALLING SEQUENCE: JSR PC,CHKINT
                                            ;
                                            ;**********************************************************************

020132' 017737  160200  000670' CHKINT::MOV     @PCSR0,ERRINT                   ;GET PCSR0 CONTENTS
020140' 032737  170000  000670'         BIT     #NPRERR!NXMERR!UNIERR!PARERR,ERRINT ;ANY INTERRUPT ERRORS SET?
020146' 001405                          BEQ     10$                             ;NO
020150' 012777  170000  160160          MOV     #NPRERR!NXMERR!UNIERR!PARERR,@PCSR0 ;CLEAR ANY ERROR INTERRUPT
020156' 000261                          SEC                                     ;INDICATE ERROR
020160' 000401                          BR      20$
020162' 000241                  10$:    CLC                                     ;INDICATE NO ERRORS
020164' 000207                  20$:    RTS     PC
```

```



                                        .SBTTL  RESET UNA

                                        ;*********************************************************************
                                        ;
                                        ;FUNCTIONAL DESCRIPTION:
                                        ;       ROUTINE TO RESET DEUNA
                                        ;
                                        ;INPUTS: NONE
                                        ;
                                        ;IMPLICIT INPUTS: NONE
                                        ;
                                        ;OUTPUTS: C BIT SET IF ERROR OCCURRED
                                        ;         C BIT CLEARED IF SUCCESS
                                        ;
                                        ;SUBORDINATE ROUTINES USED: CHKDNI, CLRDNI
                                        ;
                                        ;FUNCTIONAL SIDE EFFECTS: PSW CHANGED, LINE CLOCK IS TURNED ON, OPERATIONAL
                                        ;MICROCODE IS STARTED.
                                        ;
                                        ;CALLING SEQUENCE: JSR PC,REUNA
                                        ;
                                        ;*********************************************************************8

 020166' 012777  000040  160142  REUNA:: MOV     #RSET,@PCSR0            ;RESET DEUNA BACK TO OPERATIONAL MICRO
 020174' 012737  001364  000332'         MOV     #12.*SECOND,METER       ;PUT SOME TIME ON THE METER
 020202' 004737  017316'                 JSR     PC,CHKDNI               ;WAIT FOR 'DNI'
 020206' 103037                          BCC     10$                     ;OK
                                                                         ;ERROR DNI NOT SET AFTER RESET!
 020210' 012737  001000' 000310'         MOV     #SDNI,BITNAM            ;SETUP ERROR MESSAGE
 020216' 012737  001277' 000312'         MOV     #SNSET,BITSTA
 020224' 012737  001342' 000314'         MOV     #SAFTER,PWHEN
 020232' 005037  000304'                 CLR     CSRNUM
 020236'                                 PRINTF  #FORM9,CSRNUM,BITNAM,BITSTA,PWHEN
 020236' 013746  000314'                                                                 MOV     PWHEN,-(SP)
 020242' 013746  000312'                                                                 MOV     BITSTA,-(SP)
 020246' 013746  000310'                                                                 MOV     BITNAM,-(SP)
 020252' 013746  000304'                                                                 MOV     CSRNUM,-(SP)
 020256' 012746  004637'                                                                 MOV     #FORM9,-(SP)
 020262' 012746  000005                                                                  MOV     #5,-(SP)
 020266' 010600                                                                          MOV     SP,R0
 020270' 104417                                                                          TRAP    CSPNTF
 020272' 062706  000014                                                                  ADD     #14,SP
 020276' 004737  017764'                 JSR     PC,PRNPCR               ;PRINT PCSR'S           ;MAC001
 020302' 000261                          SEC                             ;INDICATE ERROR
 020304' 000414                          BR      20$
 020306' 004737  017362'         10$:    JSR     PC,CLRDNI               ;GO CLEAR DNI
 020312' 103011                          BCC     20$                     ;OK
 020314'                                 PRINTF  #FORM8
 020314' 012746  004555'                                                                 MOV     #FORM8,-(SP)
 020320' 012746  000001                                                                  MOV     #1,-(SP)
 020324' 010600                                                                          MOV     SP,R0
 020326' 104417                                                                          TRAP    CSPNTF
 020330' 062706  000004                                                                  ADD     #4,SP
```

```
4559  020334' 000261                    SEC
4560  020336' 000207          20$:      RTS       PC
```

```

                                          .SBTTL  LOAD MICROMODULE

                                          ;*******************************************************************
                                          ;
                                          ;THIS SUBROUTINE LOADS THE MICROMODULE FOR TESTS THAT REQUIRE CUSTOM
                                          ;MICROCODE.
                                          ;
                                          ;*******************************************************************

 020340' 012737  000326' 000320' LODMIC: MOV     #MICRO,MICMOD           ;POINT TO MICRO MODULE NAME
 020346' 004737  020166'                 JSR     PC,REUNA                ;GO START OPERATION MICROCODE
 020352' 103015                          BCC     5$                      ;OK
 020354'                                 PRINTF  #UNLOD,MICMOD           ;PRINT ERROR MESSAGE
 020354' 013746  000320'                                                         MOV     MICMOD,-(SP)
 020360' 012746  001517'                                                         MOV     #UNLOD,-(SP)
 020364' 012746  000002                                                          MOV     #2,-(SP)
 020370' 010600                                                                  MOV     SP,R0
 020372' 104417                                                                  TRAP    C$PNTF
 020374' 062706  000006                                                          ADD     #6,SP
 020400' 000261                          SEC                             ;INDICATE ERROR OCCURRED
 020402' 000137  021174'                 JMP     200$                    ;LEAVE
 020406' 023727  000326' 000101  5$:     CMP     MICRO,#'A                       ;LOAD MICROCODE MODULE A?
 020414' 001007                          BNE     10$                     ;NO
                                                                         ;YES
 020416' 013737  000000G 000616'         MOV     MICASZ,UDBB             ;SIZE OF MODULE A
 020424' 012737  000000G 000620'         MOV     #MICROA,UDBB+2          ;BASE ADDRESS OF MODULE A
 020432' 000475                          BR      70$

 020434' 023727  000326' 000102  10$:    CMP     MICRO,#'B                       ;LOAD MICROCODE MODULE B?
 020442' 001007                          BNE     20$                     ;NO
                                                                         ;YES
 020444' 013737  000000G 000616'         MOV     MICBSZ,UDBB             ;SIZE OF MODULE B
 020452' 012737  000000G 000620'         MOV     #MICROB,UDBB+2          ;BASE ADDRESS OF MODULE B
 020460' 000462                          BR      70$

 020462' 023727  000326' 000103  20$:    CMP     MICRO,#'C                       ;LOAD MICROCODE MODULE C?
 020470' 001007                          BNE     30$                     ;NO
                                                                         ;YES
 020472' 013737  000000G 000616'         MOV     MICCSZ,UDBB             ;SIZE OF MODULE C
 020500' 012737  000000G 000620'         MOV     #MICROC,UDBB+2          ;BASE ADDRESS OF MODULE C
 020506' 000447                          BR      70$

 020510' 023727  000326' 000104  30$:    CMP     MICRO,#'D                       ;LOAD MICROCODE MODULE D?
 020516' 001007                          BNE     40$                     ;NO
                                                                         ;YES
 020520' 013737  000000G 000616'         MOV     MICDSZ,UDBB             ;SIZE OF MODULE D
 020526' 012737  000000G 000620'         MOV     #MICROD,UDBB+2          ;BASE ADDRESS OF MODULE D
 020534' 000434                          BR      70$

 020536' 023727  000326' 000105  40$:    CMP     MICRO,#'E                       ;LOAD MICROCODE MODULE E?
 020544' 001007                          BNE     50$                     ;NO
                                                                         ;YES
 020546' 013737  000000G 000616'         MOV     MICESZ,UDBB             ;SIZE OF MODULE E
 020554' 012737  000000G 000620'         MOV     #MICROE,UDBB+2          ;BASE ADDRESS OF MODULE E
 020562' 000421                          BR      70$
```

```

020564' 023727  000326' 000106  50$:   CMP     MICRO,#'F                         ;LOAD MICROCODE MODULE F?
020572' 001007                         BNE     60$                       ;NO
                                                                         ;YES
020574' 013737  000000G 000616'        MOV     MICFSZ,UDBB               ;SIZE OF MODULE F
020602' 012737  000000G 000620'        MOV     #MICROF,UDBB+2            ;BASE ADDRESS OF MODULE F
020610' 000406                         BR      70$

020612' 013737  000000G 000616' 60$:   MOV     MICGSZ,UDBB               ;SIZE OF MODULE G
020620' 012737  000000G 000620'        MOV     #MICROG,UDBB+2            ;BASE ADDRESS OF MODULE G

020626' 005037  000622'         70$:   CLR     UDBB+4
020632' 012737  010000  000624'        MOV     #WCSADR+<WCSSIZ/2>,UDBB+6 ;LOAD INTO TOP HALF OF WCS

                                                                         ;SETUP PCB
020640' 012737  000021  000606'        MOV     #LIM,PCBB                 ;'LOAD INTERNAL MEMORY' FUNCTION
020646' 012737  000616' 000610'        MOV     #UDBB,PCBB+2              ;SET ADDRESS OF UDBB
020654' 005037  000612'                CLR     PCBB+4
020660' 012777  000606' 157454         MOV     #PCBB,@PCSR2              ;TELL DEUNA WHERE PCBB IS
020666' 005077  157452                 CLR     @PCSR3

020672' 004737  017410'                JSR     PC,CLRINT                 ;CLEAR ANY OUTSTANDING INTERRUPT BITS
020676' 103014                         BCC     75$                       ;OK
020700'                                PRINTF  #UNLOD,MICMOD             ;CAN'T CONTINUE WITH INTERRUPT BITS SET
020700' 013746  000320'                                                                  MOV     MICMOD,-(SP)
020704' 012746  001517'                                                                  MOV     #UNLOD,-(SP)
020710' 012746  000002                                                                   MOV     #2,-(SP)
020714' 010600                                                                           MOV     SP,R0
020716' 104417                                                                           TRAP    CSPNTF
020720' 062706  000006                                                                   ADD     #6,SP
020724' 000261                         SEC
020726' 000522                         BR      200$

020730' 012777  000001  157400  75$:   MOV     #GETPCB,@PCSR0            ;ISSUE 'GET PCB' PORT COMMAND
020736' 012737  001364  000332'        MOV     #12.*SECOND,METER         ;SETUP TIMER
020744' 004737  017316'                JSR     PC,CHKDNI                 ;WAIT FOR 'DNI' TO SET
020750' 103014                         BCC     80$                       ;OK
                                                                         ;ERROR DNI NOT SET!
020752'                                PRINTF  #UNLOD,MICMOD             ;PRINT MESSAGE
020752' 013746  000320'                                                                  MOV     MICMOD,-(SP)
020756' 012746  001517'                                                                  MOV     #UNLOD,-(SP)
020762' 012746  000002                                                                   MOV     #2,-(SP)
020766' 010600                                                                           MOV     SP,R0
020770' 104417                                                                           TRAP    CSPNTF
020772' 062706  000006                                                                   ADD     #6,SP
020776' 000261                         SEC                               ;INDICATE ERROR
021000' 000475                         BR      200$
021002'                         80$:
021002' 004737  017362'                JSR     PC,CLRDNI                 ;GO CLEAR 'DNI'
021006' 103014                         BCC     90$                       ;OK
                                                                         ;ERROR 'DNI' NOT CLEAR!
021010'                                PRINTF  #UNLOD,MICMOD             ;PRINT MESSAGE
021010' 013746  000320'                                                                  MOV     MICMOD,-(SP)
021014' 012746  001517'                                                                  MOV     #UNLOD,-(SP)
021020' 012746  000002                                                                   MOV     #2,-(SP)
021024' 010600                                                                           MOV     SP,R0
```

```
021026' 104417                                                                      TRAP    C$PNTF
021030' 062706  000006                                                              ADD     #6,SP
021034' 000261                          SEC                             ;INDICATE ERROR
021036' 000456                          BR      200$
021040'                         90$:
021040' 012777  000002  157270          MOV     #GETCMD,@PCSR0          ;ISSUE <GET COMMAND> PORT COMMAND
021046' 012737  001364  000332'         MOV     #12.*SECOND,METER       ;SETUP TIMER
021054' 004737  017316'                 JSR     PC,CHKDNI               ;WAIT FOR 'DNI'
021060' 103014                          BCC     100$                    ;OK
                                                                        ;ERROR 'DNI' NOT SET!
021062'                                 PRINTF  #UNLOD,MICMOD           ;PRINT MESSAGE
021062' 013746  000320'                                                             MOV     MICMOD,-(SP)
021066' 012746  001517'                                                             MOV     #UNLOD,-(SP)
021072' 012746  000002                                                              MOV     #2,-(SP)
021076' 010600                                                                      MOV     SP,R0
021100' 104417                                                                      TRAP    C$PNTF
021102' 062706  000006                                                              ADD     #6,SP
021106' 000261                          SEC                             ;INDICATE ERROR
021110' 000431                          BR      200$
021112'                         100$:
021112' 004737  017362'                 JSR     PC,CLRDNI               ;GO CLEAR 'DNI'
021116' 103014                          BCC     110$                    ;OK
                                                                        ;ERROR 'DNI' NOT CLEAR
021120'                                 PRINTF  #UNLOD,MICMOD           ;PRINT ERROR MESSAGE
021120' 013746  000320'                                                             MOV     MICMOD,-(SP)
021124' 012746  001517'                                                             MOV     #UNLOD,-(SP)
021130' 012746  000002                                                              MOV     #2,-(SP)
021134' 010600                                                                      MOV     SP,R0
021136' 104417                                                                      TRAP    C$PNTF
021140' 062706  000006                                                              ADD     #6,SP
021144' 000261                          SEC                             ;INDICATE ERROR
021146' 000412                          BR      200$
021150'                         110$:
021150' 012737  000001  000606'         MOV     #LASM,PCBB              ;LOAD PCBB WITH 'LOAD AND START' FUNCTION CODE
021156' 012737  010000  000610'         MOV     #WCSADR+<WCSSIZ/2>,PCBB+2 ;STARTING MICROADDRESS
021164' 012777  000002  157144          MOV     #GETCMD,@PCSR0          ;ISSUE <GET COMMAND> PORT COMMAND
021172' 000241                          CLC                             ;INDICATE SUCCESS

021174' 000207                  200$:   RTS     PC                      ;RETURN

```

```
                                .TITLE MISCELLANEOUS SECTIONS
                                .SBTTL  REPORT CODING SECTION
021176'                                 BGNRPT
021176'                                                                 LSRPT::
021176'                                 EXIT    RPT
021176' 000167                                                                  .WORD   JSJMP
021200' 000000                                                                  .WORD   L10064-2-.
021202'                                 ENDRPT
021202'                                                                 L10064:
021202' 104425                                                                  TRAP    CSRPT


                                .SBTTL  PROTECTION TABLE

                                ;++
                                ; THIS TABLE IS USED BY THE RUNTIME SERVICES
                                ; TO PROTECT THE LOAD MEDIA.
                                ;--

021204'                                 BGNPROT
021204'                                                                 LSPROT::

021204' 177777                          -1              ;OFFSET INTO P-TABLE FOR CSR ADDRESS
021206' 177777                          -1              ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
021210' 177777                          -1              ;OFFSET INTO P-TABLE FOR DRIVE NUMBER

021212'                                 ENDPROT

```

```
                                        .SBTTL  INITIALIZE SECTION

                                        ;++
                                        ; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
                                        ; AT THE BEGINNING OF EACH PASS.
                                        ;--

021212'                                        BGNINIT
021212'                                                                                 L$INIT::



021212'                                        READEF  #EF.CONTINUE            ;WAS A CONTINUE COMMAND ENTERED
021212' 012700  000036                                                                  MOV     #EF.CONTINUE,R0
021216' 104447                                                                          TRAP    C$REFG
021220'                                        BCOMPLETE       30$             ;YES, LEAVE INIT CODE
021220' 103572                                                                          BCS     30$
021222'                                        READEF  #EF.PWR                 ;WAS THERE A POWER FAILURE?
021222' 012700  000034                                                                  MOV     #EF.PWR,R0
021226' 104447                                                                          TRAP    C$REFG
                                ;MAC001 BCOMPLETE       30$             ;YES, LEAVE INIT CODE
021230'                                        BNCOMPLETE      2$              ;NO                     ;MAC001
021230' 103007                                                                          BCC     2$
                                ;                                                                       ;MAC001
                                ; DELAY A PERIOD OF TIME (APPROX 25 SECS )  FOR SELF TEST TO FINISH     ;MAC001
                                ;                                                                       ;MAC001
021232' 012701  000150                  MOV     #150,R1         ;INIT OUTER LOOP                        ;MAC001
021236' 005000                          CLR     R0              ;INIT INNER LOOP                        ;MAC001
021240' 005300                  3$:     DEC     R0                                                      ;MAC001
021242' 001376                          BNE     3$                                                      ;MAC001
021244' 005301                          DEC     R1                                                      ;MAC001
021246' 001374                          BNE     3$                                                      ;MAC001
021250'                         2$:     READEF  #EF.NEW         ;IS THIS A NEW PASS?
021250' 012700  000035                                                                  MOV     #EF.NEW,R0
021254' 104447                                                                          TRAP    C$REFG
021256'                                 BNCOMPLETE      10$     ;NO SKIP THE NEW PASS TIME STUFF
021256' 103072                                                                          BCC     10$
021260'                                 READEF  #EF.START       ;IS THIS THE VERY FIRST TIME THOUGH?
021260' 012700  000040                                                                  MOV     #EF.START,R0
021264' 104447                                                                          TRAP    C$REFG
021266'                                 BNCOMPLETE      6$      ;NO SKIP THE FIRST TIME STUFF
021266' 103061                                                                          BCC     6$
021270' 012737  000001  000674'         MOV     #1,FRSTIM       ;SET THE FIRST TIME FLAG
021276'                                 MEMORY  FRESIZ          ;GET FREE MEMORY INFO
021276' 104431                                                                          TRAP    C$MEM
021300' 010037  000322'                                                                 MOV     R0,FRESIZ
021304' 013737  000322' 000324'         MOV     FRESIZ,FREMEM   ;SIZE OF FREE MEMORY IN FRESIZ
021312' 062737  000002  000324'         ADD     #2,FREMEM       ;START OF FREE MEMORY IN FREMEM
021320'                                 CLOCK   L,R1            ;GET LINE CLOCK INFO
021320' 012700  000114                                                                  MOV     #'L,R0
021324' 104462                                                                          TRAP    C$CLCK
021326' 010001                                                                          MOV     R0,R1
021330'                                 BCOMPLETE       1$
021330' 103411                                                                          BCS     1$
021332'                                 PRINTF  #NOCLK          ;ERROR MESSAGE
021332' 012746  021610'                                                                 MOV     #NOCLK,-(SP)
```

```
021336' 012746  000001                                                                   MOV     #1,-(SP)
021342' 010600                                                                           MOV     SP,R0
021344' 104417                                                                           TRAP    C$PNTF
021346' 062706  000004                                                                   ADD     #4,SP
021352' 000512                          BR      20$                     ;CANNOT CONTINUE
021354' 012137  000274'         1$:     MOV     (R1)+,CLKCSR            ;LINE CLOCK CSR
021360' 012102                          MOV     (R1)+,R2                ;GET CLOCK PRIORITY
021362' 072227  000005                  ASH     #5,R2
021366' 010237  000276'                 MOV     R2,CLKBR
021372' 012137  000300'                 MOV     (R1)+,CLKVEC            ;VECTOR
021376' 012137  000302'                 MOV     (R1)+,CLKFRE            ;FREQUENCY
021402'                                 SETVEC  CLKVEC,#CLKSRV,CLKBR    ;SETUP CLOCK INTERRUPT VECTOR
021402' 013746  000276'                                                                  MOV     CLKBR,-(SP)
021406' 012746  022062'                                                                  MOV     #CLKSRV,-(SP)
021412' 013746  000300'                                                                  MOV     CLKVEC,-(SP)
021416' 012746  000003                                                                   MOV     #3,-(SP)
021422' 104437                                                                           TRAP    C$SVEC
021424' 062706  000010                                                                   ADD     #10,SP
021430' 000402                          BR      5$
021432' 005037  000674'         6$:     CLR     FRSTIM                  ;INDICATE NOT THE FIRST TIME THROUGH
021436' 012737  177777  000330' 5$:     MOV     #-1,UNIT                ;YES, INITIALIZE UNIT #
021444' 005237  000330'         10$:    INC     UNIT                    ;SETUP FOR NEXT UNIT
021450' 023737  000330' 000012'         CMP     UNIT,L$UNIT             ;WE TESTED ALL AVAILABLE UNITS?
021456' 003050                          BGT     20$                     ;YES, LEAVE
021460'                                 GPHARD  UNIT,R1                 ;GET P-TAB POINTER FOR THIS UNIT
021460' 013700  000330'                                                                  MOV     UNIT,R0
021464' 104442                                                                           TRAP    C$GPHRD
021466' 010001                                                                           MOV     R0,R1
021470'                                 BNCOMPLETE      10$             ;THIS ONE IS NOT AVAILABLE
021470' 103365                                                                           BCC     10$
021472' 012137  000266'                 MOV     (R1)+,UNACSR            ;SAVE CSR
021476' 012137  000270'                 MOV     (R1)+,UNAVEC            ;SAVE VECTOR
021502' 012737  000240  000272'         MOV     #PRIO5,UNAPRI           ;SAVE PRIORITY
021510' 013737  000266' 000336'         MOV     UNACSR,PCSR0            ;PCSR0
021516' 013737  000336' 000346'         MOV     PCSR0, PCSR0UB          ;PCSR0 UPPER BYTE               ;RSJ001
021524' 062737  000001  000346'         ADD     #1,PCSR0UB              ;                               ;RSJ001
021532' 013737  000336' 000340'         MOV     PCSR0,PCSR1
021540' 062737  000002  000340'         ADD     #2,PCSR1                ;PCSR1
021546' 013737  000340' 000342'         MOV     PCSR1,PCSR2
021554' 062737  000002  000342'         ADD     #2,PCSR2                ;PCSR2
021562' 013737  000342' 000344'         MOV     PCSR2,PCSR3
021570' 062737  000002  000344'         ADD     #2,PCSR3                ;PCSR3
021576' 000403                          BR      30$                     ;LEAVE
021600' 005037  000674'         20$:    CLR     FRSTIM                  ;CLEAR FIRST TIME THROUGH FLAG
021604'                                 DOCLN                           ;ABORT PASS
021604' 104444                                                                           TRAP    C$DCLN
021606'                         30$:
021606'                         ENDINIT
021606'                                                                         L10066:
021606' 104411                                                                           TRAP    C$INIT


021610' 040503  047116  052117  NOCLK:: .ASCIZ  /CANNOT CONTINUE - NEED LINE CLOCK/      ;MAC001
021616' 041440  047117  044524
021624' 052516  020105  020055
021632' 042516  042105  046040
```

```
 4853  021640' 047111  020105  046103
 4854  021646' 041517  000113
 4855                                        .EVEN
 4856
```

```
                                        .SBTTL  AUTODROP SECTION

        021652'                                 BGNAUTO
        021652'                                                                 L$AUTO::
        021652'                                 ENDAUTO
        021652'                                                                 L10067:
        021652' 104461                                                                  TRAP    C$AUTO

                                        .SBTTL  CLEANUP CODING SECTION

                                        ;++
                                        ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
                                        ; AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED.
                                        ;--
        021654'                         BGNCLN
        021654'                                                                 L$CLEAN::
        021654' 005737  000666'                 TST     NEXMEM                  ;DOES PCSR0 EXIST?
        021660' 001025                          BNE     20$                     ;NO SKIP RESET
        021662' 012777  000040  156446          MOV     #RSET,@PCSR0            ;RESTORE DEUNA TO OPERATIONAL
                                                                                ;STATE
                                        ;MAC001 MOV     #10*SECOND,METER                ;SETUP TIMER
        021670' 012737  001364  000332'         MOV     #12.*SECOND,METER       ;SET UP TIMER                   ;MAC001
        021676' 004737  017316'                 JSR     PC,CHKDNI               ;DNI SET WHEN DONE
        021702' 103012                          BCC     10$
        021704'                                 PRINTF  #CLNERR                 ;ERROR
        021704' 012746  021744'                                                                 MOV     #CLNERR,-(SP)
        021710' 012746  000001                                                                  MOV     #1,-(SP)
        021714' 010600                                                                          MOV     SP,R0
        021716' 104417                                                                          TRAP    C$PNTF
        021720' 062706  000004                                                                  ADD     #4,SP
        021724' 004737  017764'                 JSR     PC,PRNPCR               ;PRINT PCSR'S                   ;MAC001
        021730' 004737  017362'         10$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
        021734' 004737  017302'         20$:    JSR     PC,TIMOFF               ;TURN OFF THE LINE CLOCK        ;CC
        021740'                                 EXIT    CLN
        021740' 104432                                                                          TRAP    C$EXIT
        021742' 000054                                                                          .WORD   L10070-.

        021744' 047045  040445  051105  CLNERR: .ASCIZ  /%N%AERROR OCCURRED DURING DEVICE RESET%N/
        021752' 047522  020122  041517
        021760' 052503  051122  042105
        021766' 042040  051125  047111
        021774' 020107  042504  044526
        022002' 042503  051040  051505
        022010' 052105  047045  000
                022016'                         .EVEN

        022016'                         ENDCLN
        022016'                                                                 L10070:
        022016' 104412                                                                  TRAP    C$CLEAN
```

```
                                        .SBTTL  DROP UNIT SECTION

      022020'                                   BGNDU
      022020'                                                                   L$DU::
      022020'                                   EXIT    DU
      022020'  000167                                                                   .WORD   J$JMP
      022022'  000000                                                                   .WORD   L10071-2-.
      022024'                                   ENDDU
      022024'                                                                   L10071:
      022024'  104453                                                                   TRAP    C$DU

                                        .SBTTL  ADD UNIT SECTION

      022026'                                   BGNAU
      022026'                                                                   L$AU::
      022026'                                   EXIT    AU
      022026'  000167                                                                   .WORD   J$JMP
      022030'  000000                                                                   .WORD   L10072-2-.
      022032'                                   ENDAU
      022032'                                                                   L10072:
      022032'  104452                                                                   TRAP    C$AU

                                        .TITLE  GLOBAL INTERRUPT SERVICE ROUTINES

                                        .SBTTL  NON-EXISTANT MEMORY INTERRUPT SERVICE ROUTINE

                                        ;*************************************************************
                                        ;
                                        ;FUNCTIONAL DESCRIPTION:
                                        ;
                                        ;       THIS ROUTINE IS ASSIGNED TO VECTOR 4 BY THE ACCESS TESTS
                                        ;       IT SETS THE NEXMEM FLAG SIGNALING THAT AN ACCESS WAS
                                        ;       ATTEMPTED ON NON-EXISTANT MEMORY.
                                        ;
                                        ;*************************************************************

      022034'                           BGNSRV  TRAP4
      022034'                                                                   TRAP4::
      022034'  012737  000001  000666'          MOV     #1,NEXMEM               ;SET FLAG
      022042'                           ENDSRV
      022042'                                                                   L10073:
      022042'  000002                                                                   RTI
```

```
                                        .SBTTL  UNA INTERRUPT SERVICE ROUTINE

                                        ;*******************************************************************
                                        ;
                                        ;FUNCTIONAL DESCRIPTION:
                                        ;       CONTROL GOES HERE WHEN THE INTERRUPT ENABLE BIT IS SET
                                        ;       AND ANY OF THE INTERRUPT BITS SET.
                                        ;       A COPY OF PCSRO IS STORED AT 'UNAINT:' AND A COPY OF THE
                                        ;       PSW AT THE TIME OF THE INTERRUPT IS STORED AT 'CPUPRI:'.
                                        ;
                                        ;*******************************************************************

 022044'                                BGNSRV  UNASRV
 022044'                                                                            UNASRV::
 022044' 017737  156266  000672'                MOV     @PCSRO,UNAINT           ;INDICATE UNA INTERRUPTED
 022052' 016637  000002  000676'                MOV     2(SP),CPUPRI            ;GET CPU PRIORITY AT TIME OF INTERRUPT
 022060'                                ENDSRV
 022060'                                                                            L10074:
 022060' 000002                                                                             RTI


                                        .SBTTL  CLOCK INTERRUPT SERVICE ROUTINE

                                        ;*******************************************************************
                                        ;
                                        ;FUNCTIONAL DESCRIPTION:
                                        ;
                                        ;       THIS ROUTINE COUNTS A PRESET NUMBER OF CLOCK TICKS THEN IT
                                        ;       TURNS THE CLOCK OFF
                                        ;
                                        ;INPUTS: METER
                                        ;
                                        ;OUTPUTS:METER
                                        ;
                                        ;ROUTINES CALLED: NONE
                                        ;
                                        ;*******************************************************************

 022062'                                BGNSRV  CLKSRV
 022062'                                                                            CLKSRV::
 022062' 005737  000332'                        TST     METER           ;HAS THE METER EXPIRED?
 022066' 001402                                 BEQ     20$             ;YES, STOP COUNTING
 022070' 005337  000332'                        DEC     METER           ;COUNT TICKS
 022074'                                20$:
 022074'                                ENDSRV
 022074'                                                                            L10075:
 022074' 000002                                                                             RTI
```

```
                                  .TITLE HARDWARE TESTS

                                  .SBTTL  TEST 1: PCSRO READ ACCESS TEST

                                  ;*******************************************************************
                                  ;
                                  ;THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER 0
                                  ;CAN BE READ FROM THE UNIBUS AND THAT THE PREDETERMINED BITS APPEAR
                                  ;IN THE EXPECTED BIT POSITIONS.
                                  ;
                                  ;TEST SEQUENCE: 1-READ PCSRO
                                  ;               2-VERIFY BITS 9 AND 4 = 0
                                  ;
                                  ;*******************************************************************

022076'                           BGNTST
022076'                                                                  T1::
022076'                                   SETVEC  #4,#TRAP4,#PRI07               ;SETUP TIME-OUT TRAP
022076' 012746  000340                                                                  MOV     #PRI07,-(SP)
022102' 012746  022034'                                                                 MOV     #TRAP4,-(SP)
022106' 012746  000004                                                                  MOV     #4,-(SP)
022112' 012746  000003                                                                  MOV     #3,-(SP)
022116' 104437                                                                          TRAP    C$SVEC
022120' 062706  000010                                                                  ADD     #10,SP
022124'                            BGNSUB ;#1
022124'                                                                  T1.1:
022124' 104402                                                                          TRAP    C$BSUB

                                  ;
                                  ;CHECK TO SEE IF PCSRO EXISTS
                                  ;
022126' 005037  000666'                   CLR     NEXMEM                 ;CLEAR NON-EXISTANT MEMORY FLAG
022132' 005037  000304'                   CLR     CSRNUM                 ;HOLDS WHICH PCSR WE ARE DOING
022136' 005777  156174                    TST     @PCSRO                 ;DOES PSCRO EXIST?
022142' 005737  000666'                   TST     NEXMEM
022146' 001414                            BEQ     10$                    ;YES
022150'                                   ERRDF   001,RACERR,RACMG1      ;NO, PRINT FATAL ERROR MESSAGE
022150' 104455                                                                          TRAP    C$ERDF
022152' 000001                                                                          .WORD   1
022154' 001570'                                                                         .WORD   RACERR
022156' 012500'                                                                         .WORD   RACMG1
022160'                                   CKLOOP                         ;LOOP BACK FROM HERE IF ERROR
022160' 104406                                                                          TRAP    C$CLP1
022162'                                   CLRVEC  #4                     ;RELEASE TRAP 4 VECTOR
022162' 012700  000004                                                                  MOV     #4,R0
022166' 104436                                                                          TRAP    C$CVEC
022170'                                   DODU    UNIT                   ;DROP UNIT
022170' 013700  000330'                                                                 MOV     UNIT,R0
022174' 104451                                                                          TRAP    C$DODU
022176'                                   DOCLN                          ;ABORT SUB-PASS
022176' 104444                                                                          TRAP    C$DCLN
022200'                            10$:
022200'                             ENDSUB ;#1
022200'                                                                  L10077:
022200' 104403                                                                          TRAP    C$ESUB
022202'                             BGNSUB ;#2
022202'                                                                  T1.2:
```

```
022202' 104402                                                                       TRAP    C$BSUB
                                      ;
                                      ;OK, PCSRO EXISTS NOW CHECK SOME BITS
                                      ;
022204'                                BGNSEG
022204' 104404                                                                       TRAP    C$BSEG
                                      ;
                                      ;CHECK BIT 4 = 0
                                      ;
022206' 012737  001313' 000312'             MOV     #$NCLR,BITSTA           ;TESTING FOR CLEARED BITS
022214' 012737  000004  000306'             MOV     #4,BITNUM               ;TESTING BIT 4
022222' 032777  000020  156106              BIT     #BIT4,@PCSRO            ;IS BIT 4=0?
022230' 001404                              BEQ     20$                     ;YES
022232'                                     ERRHRD  002,RACERR,RACMG2       ;NO,PRINT HARD ERROR MESSAGE
022232' 104456                                                                       TRAP    C$ERHRD
022234' 000002                                                                       .WORD   2
022236' 001570'                                                                      .WORD   RACERR
022240' 012526'                                                                      .WORD   RACMG2
022242'                               20$:
022242'                                 ENDSEG
022242'                                                                     10000$:
022242' 104405                                                                       TRAP    C$ESEG
022244'                                 BGNSEG
022244' 104404                                                                       TRAP    C$BSEG
                                      ;
                                      ;CHECK BIT 9 = 0
                                      ;
022246' 012737  000011  000306'             MOV     #9.,BITNUM              ;TESTING BIT 9 NOW
022254' 032777  001000  156054              BIT     #BIT9,@PCSRO            ;IS BIT 9=0?
022262' 001404                              BEQ     30$                     ;YES
022264'                                     ERRHRD  003,RACERR,RACMG2       ;NO,PRINT HARD ERROR MESSAGE
022264' 104456                                                                       TRAP    C$ERHRD
022266' 000003                                                                       .WORD   3
022270' 001570'                                                                      .WORD   RACERR
022272' 012526'                                                                      .WORD   RACMG2
022274'                               30$:
022274'                                 ENDSEG
022274'                                                                     10001$:
022274' 104405                                                                       TRAP    C$ESEG
022276'                                ENDSUB ;#2
022276'                                                                     L10100:
022276' 104403                                                                       TRAP    C$ESUB
022300'                                     CLRVEC  #4                      ;FREE TRAP 4 VECTOR
022300' 012700  000004                                                               MOV     #4,R0
022304' 104436                                                                       TRAP    C$CVEC
022306'                               ENDTST
022306'                                                                     L10076:
022306' 104401                                                                       TRAP    C$ETST
```

```
                                        .SBTTL  TEST 2: PCSR1 READ ACCESS TEST

                                        ;*****************************************************
                                        ;
                                        ;THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER
                                        ;CAN BE READ FROM THE UNIBUS AND THAT THE PREDETERMINED BITS
                                        ;APPEAR IN THE EXPECTED BIT POSITIONS.
                                        ;
                                        ;TEST SEQUENCE: 1-READ PCSR1
                                        ;               2-VERIFY BITS 4,5,6,14,15 = 0
                                        ;
                                        ;*****************************************************

022310'                                 BGNTST
022310'                                                                         T2::
022310'                                         SETVEC  #4,#TRAP4,#PRI07        ;SETUP TIMEOUT TRAP
022310' 012746  000340                                                                  MOV     #PRI07,-(SP)
022314' 012746  022034'                                                                 MOV     #TRAP4,-(SP)
022320' 012746  000004                                                                  MOV     #4,-(SP)
022324' 012746  000003                                                                  MOV     #3,-(SP)
022330' 104437                                                                          TRAP    C$SVEC
022332' 062706  000010                                                                  ADD     #10,SP
022336'                                  BGNSUB ;#1
022336'                                                                         T2.1:
022336' 104402                                                                          TRAP    C$BSUB

                                        ;
                                        ;CHECK TO SEE IF PCSR1 EXISTS
                                        ;
022340' 005037  000666'                         CLR     NEXMEM          ;CLEAR NON-EXISTANT MEMORY FLAG
022344' 012737  000001  000304'                 MOV     #1,CSRNUM       ;TESTING PCSR1
022352' 005777  155762                          TST     @PCSR1          ;DOES PCSR1 EXIST?
022356' 005737  000666'                         TST     NEXMEM
022362' 001414                                  BEQ     10$             ;YES
022364'                                         ERRDF   004,RACERR,RACMG1       ;NO,PRINT FATAL ERROR MESSAGE
022364' 104455                                                                          TRAP    C$ERDF
022366' 000004                                                                          .WORD   4
022370' 001570'                                                                         .WORD   RACERR
022372' 012500'                                                                         .WORD   RACMG1
022374'                                         CKLOOP                  ;LOOP FROM HERE IF ERROR
022374' 104406                                                                          TRAP    C$CLP1
022376'                                         CLRVEC  #4              ;RELEASE TRAP 4 VECTOR
022376' 012700  000004                                                                  MOV     #4,R0
022402' 104436                                                                          TRAP    C$CVEC
022404'                                         DODU    UNIT            ;DROP THE UNIT
022404' 013700  000330'                                                                 MOV     UNIT,R0
022410' 104451                                                                          TRAP    C$DODU
022412'                                         DOCLN                   ;ABORT SUB-PASS
022412' 104444                                                                          TRAP    C$DCLN
022414'                                  10$:
022414'                                   ENDSUB ;#1
022414'                                                                         L10102:
022414' 104403                                                                          TRAP    C$ESUB
022416'                                   BGNSUB ;#2
022416'                                                                         T2.2:
022416' 104402                                                                          TRAP    C$BSUB
```

```
                                        ;
                                        ;OK, PCSR1 EXISTS NOW CHECK SOME BITS
                                        ;
022420'                                   BGNSEG
022420' 104404                                                                                  TRAP    C$BSEG
                                        ;
                                        ;CHECK BIT 4 = 0
                                        ;
022422' 012737  001313' 000312'                 MOV     #$NCLR,BITSTA           ;TESTING CLEARED BITS
022430' 012737  000004  000306'                 MOV     #4,BITNUM               ;TESTING BIT 4
022436' 032777  000020  155674                  BIT     #BIT4,@PCSR1            ;IS BIT 4=0?
022444' 001404                                  BEQ     20$                     ;YES
022446'                                         ERRHRD  005,RACERR,RACMG2       ;NO, PRINT HARD ERROR MESSAGE
022446' 104456                                                                                  TRAP    C$ERHRD
022450' 000005                                                                                  .WORD   5
022452' 001570'                                                                                 .WORD   RACERR
022454' 012526'                                                                                 .WORD   RACMG2
022456'                                 20$:
022456'                                   ENDSEG
022456'                                                                                 10000$:
022456' 104405                                                                                  TRAP    C$ESEG
022460'                                   BGNSEG
022460' 104404                                                                                  TRAP    C$BSEG
                                        ;
                                        ;CHECK BIT 5 = 0
                                        ;
022462' 012737  000005  000306'                 MOV     #5,BITNUM               ;TESTING BIT 5
022470' 032777  000040  155642                  BIT     #BIT5,@PCSR1            ;IS BIT 5=0?
022476' 001404                                  BEQ     30$                     ;YES
022500'                                         ERRHRD  006,RACERR,RACMG2       ;NO,PRINT HARD ERROR MESSAGE
022500' 104456                                                                                  TRAP    C$ERHRD
022502' 000006                                                                                  .WORD   6
022504' 001570'                                                                                 .WORD   RACERR
022506' 012526'                                                                                 .WORD   RACMG2
022510'                                 30$:
022510'                                   ENDSEG
022510'                                                                                 10001$:
022510' 104405                                                                                  TRAP    C$ESEG
022512'                                   BGNSEG
022512' 104404                                                                                  TRAP    C$BSEG
                                        ;
                                        ;CHECK BIT 6 = 0
                                        ;
022514' 012737  000006  000306'                 MOV     #6,BITNUM               ;TESTING BIT 6
022522' 032777  000100  155610                  BIT     #BIT6,@PCSR1            ;IS BIT 6=0?
022530' 001404                                  BEQ     40$                     ;YES
022532'                                         ERRHRD  007,RACERR,RACMG2       ;NO,PRINT HARD ERROR MESSAGE
022532' 104456                                                                                  TRAP    C$ERHRD
022534' 000007                                                                                  .WORD   7
022536' 001570'                                                                                 .WORD   RACERR
022540' 012526'                                                                                 .WORD   RACMG2
022542'                                 40$:
022542'                                   ENDSEG
022542'                                                                                 10002$:
022542' 104405                                                                                  TRAP    C$ESEG
022544'                                   BGNSEG
```

```
022544' 104404                                                                   TRAP    C$BSEG
                                  ;
                                  ;CHECK BIT 14 = 0
                                  ;
022546' 012737  000016  000306'           MOV     #14.,BITNUM              ;TESTING BIT 14
022554' 032777  040000  155556            BIT     #BIT14,@PCSR1            ;IS BIT 14 =0?
022562' 001404                            BEQ     50$                      ;YES
022564'                                   ERRHRD  008,RACERR,RACMG2        ;NO, PRINT ERROR MESSAGE
022564' 104456                                                                   TRAP    C$ERHRD
022566' 000010                                                                   .WORD   8
022570' 001570'                                                                  .WORD   RACERR
022572' 012526'                                                                  .WORD   RACMG2
022574'                           50$:
022574'                             ENDSEG
022574'                                                                  10003$:
022574' 104405                                                                   TRAP    C$ESEG
022576'                             BGNSEG
022576' 104404                                                                   TRAP    C$BSEG
                                  ;
                                  ;CHECK BIT 15 = 0
                                  ;
022600' 012737  000017  000306'           MOV     #15.,BITNUM              ;TESTING BIT 15
022606' 032777  100000  155524            BIT     #BIT15,@PCSR1            ;IS BIT 15 = 0?
022614' 001404                            BEQ     60$                      ;YES
022616'                                   ERRHRD  009,RACERR,RACMG2        ;NO, PRINT ERROR MESSAGE
022616' 104456                                                                   TRAP    C$ERHRD
022620' 000011                                                                   .WORD   9
022622' 001570'                                                                  .WORD   RACERR
022624' 012526'                                                                  .WORD   RACMG2
022626'                           60$:
022626'                             ENDSEG
022626'                                                                  10004$:
022626' 104405                                                                   TRAP    C$ESEG
022630'                            ENDSUB ;#2
022630'                                                                  L10103:
022630' 104403                                                                   TRAP    C$ESUB
022632'                                   CLRVEC  #4                       ;FREE VECTOR 4
022632' 012700  000004                                                           MOV     #4,R0
022636' 104436                                                                   TRAP    C$CVEC
022640'                           ENDTST
022640'                                                                  L10101:
022640' 104401                                                                   TRAP    C$ETST
```

```
                                .SBTTL  TEST 3: PCSR2 READ ACCESS TEST

                                ;*******************************************************************
                                ;
                                ;THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER 2
                                ;CAN BE READ FROM THE UNIBUS
                                ;
                                ;TEST SEQUENCE: 1-READ PCSR2
                                ;
                                ;*******************************************************************

022642'                         BGNTST
022642'                                                                         T3::
022642'                                 SETVEC  #4,#TRAP4,PRI07         ;SETUP TIMEOUT TRAP VECTOR
022642' 013746  000340                                                                  MOV     PRI07,-(SP)
022646' 012746  022034'                                                                 MOV     #TRAP4,-(SP)
022652' 012746  000004                                                                  MOV     #4,-(SP)
022656' 012746  000003                                                                  MOV     #3,-(SP)
022662' 104437                                                                          TRAP    C$SVEC
022664' 062706  000010                                                                  ADD     #10,SP
022670' 005037  000666'                 CLR     NEXMEM                  ;CLEAR NON-EXISTANT MEMORY FLAG
022674' 012737  000002  000304'         MOV     #2,CSRNUM               ;TESTING PCSR2
022702' 005777  155434                  TST     @PCSR2                  ;DOES PCSR2 EXIST?
022706' 005737  000666'                 TST     NEXMEM
022712' 001414                          BEQ     10$                     ;YES
022714'                                 ERRDF   010,RACERR,RACMG1       ;NO,PRINT FATAL ERROR MESSAGE
022714' 104455                                                                          TRAP    C$ERDF
022716' 000012                                                                          .WORD   10
022720' 001570'                                                                         .WORD   RACERR
022722' 012500'                                                                         .WORD   RACMG1
022724'                                 CKLOOP                          ;LOOP BACK FROM HERE IF ERROR
022724' 104406                                                                          TRAP    C$CLP1
022726'                                 CLRVEC  #4                      ;RELEASE TRAP 4 VECTOR
022726' 012700  000004                                                                  MOV     #4,R0
022732' 104436                                                                          TRAP    C$CVEC
022734'                                 DODU    UNIT                    ;DROP THE UNIT
022734' 013700  000330'                                                                 MOV     UNIT,R0
022740' 104451                                                                          TRAP    C$DODU
022742'                                 DOCLN                           ;ABORT SUBPASS
022742' 104444                                                                          TRAP    C$DCLN
022744'                         10$:    CLRVEC  #4                      ;RELEASE TRAP 4 VECTOR
022744' 012700  000004                                                                  MOV     #4,R0
022750' 104436                                                                          TRAP    C$CVEC

022752'                         ENDTST
022752'                                                                         L10104:
022752' 104401                                                                          TRAP    C$ETST
```

```
                                        .SBTTL  TEST 4: PCSR3 READ ACCESS TEST

                                        ;*******************************************************************
                                        ;
                                        ;THIS TEST WILL VERIFY THAT THE PORT CONTROL AND STATUS REGISTER 3
                                        ;CAN BE READ FROM THE UNIBUS
                                        ;
                                        ;TEST SEQUENCE: 1-READ PCSR3
                                        ;
                                        ;*******************************************************************

022754'                                 BGNTST
022754'                                                                         T4::
022754'                                         SETVEC  #4,#TRAP4,#PRI07        ;SETUP VECTOR FOR TIME-OUT
022754' 012746  000340                                                                  MOV     #PRI07,-(SP)
022760' 012746  022034'                                                                 MOV     #TRAP4,-(SP)
022764' 012746  000004                                                                  MOV     #4,-(SP)
022770' 012746  000003                                                                  MOV     #3,-(SP)
022774' 104437                                                                          TRAP    C$SVEC
022776' 062706  000010                                                                  ADD     #10,SP
023002'                                   BGNSEG
023002' 104404                                                                          TRAP    C$BSEG
023004' 005037  000666'                         CLR     NEXMEM                  ;CLEAR NON-EXISTANT MEMORY FLAG
023010' 012737  000003  000304'                 MOV     #3,CSRNUM               ;TESTING PCSR3
023016' 005777  155322                          TST     @PCSR3                  ;DOES PCSR3 EXIST?
023022' 005737  000666'                         TST     NEXMEM
023026' 001414                                  BEQ     10$                     ;YES
023030'                                         ERRDF   011,RACERR,RACMG1       ;NO,PRINT FATAL ERROR MESSAGE
023030' 104455                                                                          TRAP    C$ERDF
023032' 000013                                                                          .WORD   11
023034' 001570'                                                                         .WORD   RACERR
023036' 012500'                                                                         .WORD   RACMG1
023040'                                         CKLOOP                          ;LOOP BACK FROM HERE IF ERROR
023040' 104406                                                                          TRAP    C$CLP1
023042'                                         CLRVEC  #4                      ;RELEASE TRAP 4 VECTOR
023042' 012700  000004                                                                  MOV     #4,R0
023046' 104436                                                                          TRAP    C$CVEC
023050'                                         DODU    UNIT                    ;DROP UNIT
023050' 013700  000330'                                                                 MOV     UNIT,R0
023054' 104451                                                                          TRAP    C$DODU
023056'                                         DOCLN                           ;ABORT SUB-PASS
023056' 104444                                                                          TRAP    C$DCLN
023060'                                 10$:
023060'                                   ENDSEG
023060'                                                                         10000$:
023060' 104405                                                                          TRAP    C$ESEG
023062'                                         CLRVEC  #4                      ;RETURN TIME-OUT TRAP
023062' 012700  000004                                                                  MOV     #4,R0
023066' 104436                                                                          TRAP    C$CVEC
023070'                                 ENDTST
023070'                                                                         L10105:
023070' 104401                                                                          TRAP    C$ETST
```

```
                                                  .SBTTL  TEST 5: RESET TEST

                                                  ;***************************************************************
                                                  ;
                                                  ;THIS TEST WILL VERIFY THE RESET STATE FOR ALL DEUNA UNIBUS REGISTERS
                                                  ;
                                                  ;TEST SEQUENCE:
                                                  ;       1-WRITE A 1 TO PCSR0 BIT 5
                                                  ;       2-READ PCSR0
                                                  ;        -VERIFY DNI SET
                                                  ;        -VERIFY INTR SET
                                                  ;       3-VERIFY PCSR0 BITS 15:12 AND 10:08 AND 06:04 FOR LOGICAL 0     ;MAC001
                                                  ;       4-READ PCSR1
                                                  ;       5-VERIFY PCSR1 BITS 07:03 AND 14 AND 15 FOR LOGICAL 0 AND
                                                  ;         PORT STATE FIELD BITS 02:00 FOR READY STATE
                                                  ;
                                                  ;***************************************************************

 023072'                                          BGNTST
 023072'                                                                                    T5::
 023072' 005037  000304'                                  CLR     CSRNUM                  ;TESTING PCSR0 FIRST
 023076' 012737  001342' 000314'                          MOV     #$AFTER,PWHEN           ;SETUP PART OF ERROR MESSAGE
                                                                                          ;FOR THE FOLLOWING TEST SEGMENTS
 023104'                                           BGNSUB ;#1
 023104'                                                                                    T5.1:
 023104' 104402                                                                                          TRAP    C$BSUB
                                                  ;
                                                  ;SET THE RESET BIT IN PCSR0 AND WAIT FOR 'DNI' TO SET
                                                  ;
 023106' 012777  000040  155222                           MOV     #RSET,@PCSR0            ;RESET DEUNA
                                                  ;MAC001 MOV     #10*SECOND,METER        ;PUT SOME TIME ON THE METER
 023114' 012737  001364  000332'                          MOV     #12.*SECOND,METER       ;PUT PLENTY OF TIME ON THE METER;MAC001
 023122' 004737  017316'                                  JSR     PC,CHKDNI               ;WAIT FOR DNI TO SET
 023126' 103014                                           BCC     10$                     ;OK
                                                                                          ;ERROR 'DNI' DID NOT SET
 023130' 012737  001000' 000310'                          MOV     #$DNI,BITNAM            ;SETUP ERROR MESSAGE
 023136' 012737  001277' 000312'                          MOV     #$NSET,BITSTA
 023144'                                                  ERRHRD  012,RSETER,MSG3         ;PRINT ERROR MESSAGE
 023144' 104456                                                                                          TRAP    C$ERHRD
 023146' 000014                                                                                          .WORD   12
 023150' 001624'                                                                                         .WORD   RSETER
 023152' 013030'                                                                                         .WORD   MSG3
 023154'                                                  ESCAPE  TST                     ;NO POINT IN CONTINUING TEST
 023154' 104410                                                                                          TRAP    C$ESCAPE
 023156' 000400                                                                                          .WORD   L10106-.
 023160'                                          10$:
 023160'                                           ENDSUB ;#1
 023160'                                                                                    L10107:
 023160' 104403                                                                                          TRAP    C$ESUB

 023162'                                           BGNSUB ;#2
 023162'                                                                                    T5.2:
 023162' 104402                                                                                          TRAP    C$BSUB
                                                  ;
                                                  ;CHECK ALL THE BITS AFFECTED BY RESET
                                                  ;
```

```
023164'                                 BGNSEG
023164' 104404                                                                          TRAP    C$BSEG
                                ;
                                ;CHECK THE INTERRUPT SUMMARY BIT = 0
                                ;
023166' 032777  000200  155142          BIT     #INTR,@PCSR0            ;IS THE INTERRUPT SUMMARY BIT SET?
023174' 001012                          BNE     11$                     ;YES
                                                                        ;ERROR, INTERRUPT SUMMARY NOT SET!
023176' 012737  001032' 000310'         MOV     #$INTR,BITNAM           ;PREPARE ERROR MESSAGE
023204' 012737  001277' 000312'         MOV     #$NSET,BITSTA
023212'                                 ERRHRD  013,RSETER,MSG3         ;PRINT ERROR MESSAGE
023212' 104456                                                                          TRAP    C$ERHRD
023214' 000015                                                                          .WORD   13
023216' 001624'                                                                         .WORD   RSETER
023220' 013030'                                                                         .WORD   MSG3
023222'                         11$:
023222'                           ENDSEG
023222'                                                                         10000$:
023222' 104405                                                                          TRAP    C$ESEG
                                ;
                                ;CHECK BITS 15:12 AND 10:08 AND 06:04 OF PCSR0 FOR 0                    ;MAC001
                                ;
023224' 012701  000360'                 MOV     #BNAMT0,R1              ;POINT TO BIT NAME MNEMONICS
023230' 062701  000010                  ADD     #4*2,R1                 ;INDEX DOWN TO BIT 4 MNEMONIC   ;MAC001
                                ;MAC001 MOV     #BIT0,R2                ;START TESTING AT LSB
023234' 012702  000020                  MOV     #BIT4,R2                ;START TESTING AT BIT 4         ;MAC001
023240' 012737  001313' 000312'         MOV     #$NCLR,BITSTA           ;TESTING FOR CLEARED BITS
023246'                         15$:
023246'                           BGNSEG
023246' 104404                                                                          TRAP    C$BSEG
023250' 030277  155062                  BIT     R2,@PCSR0               ;IS THIS BIT CLEARED?
023254' 001406                          BEQ     20$                     ;YES
                                                                        ;ERROR BIT FAILED TO CLEAR
023256' 011137  000310'                 MOV     (R1),BITNAM             ;GET MNEMONIC FOR THIS BIT
023262'                                 ERRHRD  014,RSETER,MSG3         ;PRINT ERROR MESSAGE
023262' 104456                                                                          TRAP    C$ERHRD
023264' 000016                                                                          .WORD   14
023266' 001624'                                                                         .WORD   RSETER
023270' 013030'                                                                         .WORD   MSG3
023272'                         20$:
023272'                           ENDSEG
023272'                                                                         10001$:
023272' 104405                                                                          TRAP    C$ESEG
023274' 005702                  21$:    TST     R2                      ;WE TESTED ALL BITS IN PCSR0?
023276' 100411                          BMI     25$                     ;YES
023300' 006302                          ASL     R2                      ;POINT TO NEXT BIT TO TEST
023302' 062701  000002                  ADD     #2,R1                   ;POINT TO NEXT MNEMONIC
023306' 105702                          TSTB    R2                      ;ARE WE POINTING TO 'INTR' BIT?
023310' 100771                          BMI     21$                     ;YES, SKIP IT
023312' 022702  004000                  CMP     #DNI,R2                 ;ARE WE POINTING TO 'DNI' BIT?
023316' 001766                          BEQ     21$                     ;YES SKIP IT
023320' 000752                          BR      15$                     ;CONTINUE TESTING

023322'                         25$:
023322'                           BGNSEG
023322' 104404                                                                          TRAP    C$BSEG
```

```
                                        ;
                                        ;CHECK PCSR1 BITS 02:00 FOR THE READY STATE
                                        ;
023324' 012737  000001  000304'                 MOV     #1,CSRNUM               ;TESTING PCSR1
023332' 017701  155002                          MOV     @PCSR1,R1               ;GET PCSR1 CONTENTS
023336' 042701  177770                          BIC     #^CPSTATE,R1            ;MASK ALL BUT PORT STATUS BITS
023342' 022701  000002                          CMP     #READY,R1               ;IS THE DEUNA IN THE READY STATE?
023346' 001404                                  BEQ     30$                     ;YES
                                                                                ;ERROR, DEUNA NOT IN READY STATE!
023350'                                         ERRHRD  015,RSETER,MSG4         ;PRINT ERROR MESSAGE
023350' 104456                                                                          TRAP    C$ERHRD
023352' 000017                                                                          .WORD   15
023354' 001624'                                                                         .WORD   RSETER
023356' 013076'                                                                         .WORD   MSG4
023360'                                 30$:
023360'                                   ENDSEG
023360'                                                                         10002$:
023360' 104405                                                                          TRAP    C$ESEG
                                        ;
                                        ;CHECK PCSR1 BITS 07:03 FOR 0
                                        ;
023362' 012701  000420'                         MOV     #BNAMT1,R1              ;POINT TO MNEMONIC TABLE FOR PCSR1
023366' 062701  000006                          ADD     #3*2,R1                 ;INDEX PAST STATE BITS
023372' 012737  001313' 000312'                 MOV     #$NCLR,BITSTA           ;PREPARE ERROR MESSAGE
023400' 012737  001342' 000314'                 MOV     #$AFTER,PWHEN
023406' 012702  000010                          MOV     #BIT3,R2                ;POINT TO BIT 3 TO START TESTING
023412'                                 40$:
023412'                                   BGNSEG
023412' 104404                                                                          TRAP    C$BSEG
023414' 030277  154720                          BIT     R2,@PCSR1               ;IS THIS BIT A 0?
023420' 001406                                  BEQ     45$                     ;YES
                                                                                ;ERROR, BIT IS NOT A 0 AFTER RESET!
023422' 011137  000310'                         MOV     (R1),BITNAM             ;GET MNEMONIC FOR THIS BIT
023426'                                         ERRHRD  016,RSETER,MSG3         ;PRINT ERROR MESSAGE
023426' 104456                                                                          TRAP    C$ERHRD
023430' 000020                                                                          .WORD   16
023432' 001624'                                                                         .WORD   RSETER
023434' 013030'                                                                         .WORD   MSG3
023436'                                 45$:
023436'                                   ENDSEG
023436'                                                                         10003$:
023436' 104405                                                                          TRAP    C$ESEG
023440' 105702                                  TSTB    R2                      ;WE TESTED BITS 07:03?
023442' 100404                                  BMI     50$                     ;YES
023444' 006302                                  ASL     R2                      ;NO, POINT TO NEXT BIT
023446' 062701  000002                          ADD     #2,R1                   ;POINT TO MNEMONIC FOR NEXT BIT
023452' 000757                                  BR      40$                     ;CONTINUE TESTING
023454'                                 50$:
023454'                                   BGNSEG
023454' 104404                                                                          TRAP    C$BSEG
                                        ;
                                        ;CHECK PCSR1 BIT 14 = 0
                                        ;
023456' 032777  040000  154654                  BIT     #ICAB,@PCSR1            ;IS PORT/CABLING BIT CLEAR?
023464' 001412                                  BEQ     60$                     ;YES
                                                                                ;ERROR, PORT/CABLING BIT NOT CLEAR!
```

```
023466' 012737  001313' 000312'         MOV     #$NCLR,BITSTA           ;PREPARE ERROR MESSAGE
023474' 012737  001076' 000310'         MOV     #$ICAB,BITNAM           ;GET MNEMONIC
023502'                                 ERRHRD  017,RSETER,MSG3         ;PRINT ERROR MESSAGE
023502' 104456                                                                          TRAP    C$ERHRD
023504' 000021                                                                          .WORD   17
023506' 001624'                                                                         .WORD   RSETER
023510' 013030'                                                                         .WORD   MSG3
023512'                         60$:
023512'                           ENDSEG
023512'                                                                         10004$:
023512' 104405                                                                          TRAP    C$ESEG
023514'                           BGNSEG
023514' 104404                                                                          TRAP    C$BSEG
                                ;
                                ;CHECK PCSR1 BIT 15 = 0
                                ;
023516' 032777  100000  154614          BIT     #XPWR,@PCSR1            ;IS TRANSCEIVER POWER BIT CLEAR?
023524' 001412                          BEQ     70$                     ;YES
                                                                        ;ERROR, TRANSCEIVER POWER BIT NOT CLEAR!
023526' 012737  001313' 000312'         MOV     #$NCLR,BITSTA           ;PREPARE ERROR MESSAGE
023534' 012737  001065' 000310'         MOV     #$XPWR,BITNAM           ;GET MNEMONIC
023542'                                 ERRHRD  018,RSETER,MSG3         ;PRINT ERROR MESSAGE
023542' 104456                                                                          TRAP    C$ERHRD
023544' 000022                                                                          .WORD   18
023546' 001624'                                                                         .WORD   RSETER
023550' 013030'                                                                         .WORD   MSG3
023552'                         70$:
023552'                           ENDSEG
023552'                                                                         10005$:
023552' 104405                                                                          TRAP    C$ESEG
023554'                          ENDSUB ;#2
023554'                                                                         L10110:
023554' 104403                                                                          TRAP    C$ESUB
023556'                         ENDTST
023556'                                                                         L10106:
023556' 104401                                                                          TRAP    C$ETST
```

```
                                        .SBTTL  TEST 6: PCSR2 REGISTER READ/WRITE TEST

                                        ;**********************************************************************
                                        ;
                                        ;THIS TEST WILL CHECK THE REGISTER FOR ALL SA0 AND SA1 ERRORS (STUCK AT
                                        ;0 AND STUCK AT 1 ERRORS). THE HOST WILL WRITE PATTERNS TO THE REGISTER
                                        ;AND READ THEM BACK TO VERIFY. THE PATTERNS TO BE USED ARE AT THE LABEL
                                        ;PATERN:: IN THE GLOBAL DATA SECTION OF THIS PROGRAM.
                                        ;
                                        ;NOTE: SINCE PCSR2 BIT 00 IS ALWAYS PRESET TO LOGIC 0, THE LOWEST ORDER
                                        ;BIT OF THE PATTERN WILL BE MASKED BEFORE DOING THE COMPARISON.
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-WRITE PATTERN TO REGISTER PCSR2
                                        ;       2-COMPARE MASKED PATTERN WITH REGISTER PCSR2 CONTENTS
                                        ;       3-REPEAT STEPS 1 TO 2 FOR ALL PATTERNS
                                        ;
                                        ;**********************************************************************

023560'                                 BGNTST
023560'                                                                                 T6::
023560' 012737  000002  000304'                 MOV     #2,CSRNUM               ;TESTING PCSR2
023566' 012701  000520'                         MOV     #PATERN,R1              ;GET ADDRESS OF DATA PATTERNS
023572' 012705  000005                          MOV     #5,R5                   ;5 DATA PATTERNS
023576' 012103                          10$:    MOV     (R1)+,R3                ;GET DATA PATTERN
023600'                                   BGNSEG
023600' 104404                                                                          TRAP    C$BSEG
023602' 010377  154534                          MOV     R3,@PCSR2               ;WRITE PATTERN TO PCSR2
023606' 017704  154530                          MOV     @PCSR2,R4               ;READ PCSR2
023612' 042703  000001                          BIC     #BIT0,R3                ;MASK BIT 00
023616' 020304                                  CMP     R3,R4                   ;COMPARE WHAT WAS WRITTEN TO...
                                                                                ;WHAT WAS READ
023620' 001404                                  BEQ     20$                     ;COMPARED OK
023622'                                         ERRHRD  019,RRWER,RACMG3        ;PRINT ERROR MESSAGE
023622' 104456                                                                          TRAP    C$ERHRD
023624' 000023                                                                          .WORD   19
023626' 001646'                                                                         .WORD   RRWER
023630' 012570'                                                                         .WORD   RACMG3
023632'                                 20$:
023632'                                   ENDSEG
023632'                                                                                 10000$:
023632' 104405                                                                          TRAP    C$ESEG
023634' 005305                                  DEC     R5                      ;DONE ALL DATA PATTERNS?
023636' 001357                                  BNE     10$                     ;NO, CONTINUE TESTING
023640'                                 ENDTST
023640'                                                                                 L10111:
023640' 104401                                                                          TRAP    C$ETST
```

```
                                            .SBTTL  TEST 7: REGISTER PCSR3 READ/WRITE TEST

                                            ;*****************************************************************
                                            ;
                                            ;THIS TEST WILL WRITE PATTERNS TO THE WRITEABLE FIELD OF PCSR3
                                            ;AND WILL READ THESE BACK FOR VERIFICATION.
                                            ;
                                            ;TEST SEQUENCE:
                                            ;       1-WRITE PATTERN 000001 TO PCSR3
                                            ;       2-VERIFY PATTERN IN PCSR3
                                            ;       3-WRITE PATTERN 000002 TO PCSR3
                                            ;       4-VERIFY PATTERN IN PCSR3
                                            ;
                                            ;*******************************************************************

 023642'                                    BGNTST
 023642'                                                                               T7::
 023642' 012737  000003  000304'                   MOV     #3,CSRNUM               ;TESTING PCSR3
 023650' 012703  000001                            MOV     #BIT00,R3               ;DATA PATTERN =1
 023654'                                      BGNSEG
 023654' 104404                                                                                   TRAP    C$BSEG
 023656' 010377  154462                            MOV     R3,@PCSR3               ;WRITE PATTERN TO PCSR3
 023662' 017704  154456                            MOV     @PCSR3,R4               ;READ PCSR3
 023666' 020304                                    CMP     R3,R4                   ;COMPARE
 023670' 001404                                    BEQ     10$                     ;IF == GOOD
 023672'                                           ERRHRD  020,RRWER,RACMG3        ;ELSE PRINT ERROR MESSAGE
 023672' 104456                                                                                   TRAP    C$ERHRD
 023674' 000024                                                                                   .WORD   20
 023676' 001646'                                                                                  .WORD   RRWER
 023700' 012570'                                                                                  .WORD   RACMG3
 023702'                                    10$:
 023702'                                      ENDSEG
 023702'                                                                               10000$:
 023702' 104405                                                                                   TRAP    C$ESEG
 023704' 006303                                    ASL     R3                      ;DATA PATTERN =2
 023706'                                      BGNSEG
 023706' 104404                                                                                   TRAP    C$BSEG
 023710' 010377  154430                            MOV     R3,@PCSR3               ;WRITE PATTERN TO PCSR3
 023714' 017704  154424                            MOV     @PCSR3,R4               ;READ PCSR3
 023720' 020304                                    CMP     R3,R4                   ;COMPARE
 023722' 001404                                    BEQ     20$                     ;IF = GOOD
 023724'                                           ERRHRD  021,RRWER,RACMG3        ;ELSE PRINT ERROR MESSAGE
 023724' 104456                                                                                   TRAP    C$ERHRD
 023726' 000025                                                                                   .WORD   21
 023730' 001646'                                                                                  .WORD   RRWER
 023732' 012570'                                                                                  .WORD   RACMG3
 023734'                                    20$:
 023734'                                      ENDSEG
 023734'                                                                               10001$:
 023734' 104405                                                                                   TRAP    C$ESEG
 023736'                                    ENDTST
 023736'                                                                               L10112:
 023736' 104401                                                                                   TRAP    C$ETST
```

```
                                        .SBTTL  TEST 8: NOP TEST

                                        ;*****************************************************************
                                        ;
                                        ;THIS TEST WILL VERIFY THAT THE DEUNA PROCESSOR IS ALIVE AND CAN
                                        ;RESPOND TO A PORT COMMAND ISSUED. THE NOP PORT COMMAND WILL BE ISSUED
                                        ;TO THE DEUNA IN PCSRO BITS 3:0 AND WILL WAIT FOR THE 'DNI' BIT TO
                                        ;SET IN PCSRO.
                                        ;
                                        ;THE NOP PORT COMMAND USES A MINIMUM OF HARDWARE BUT FORCES THE T11
                                        ;TO EXECUTE THE PORT COMMAND SEQUENCE.
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-READ REGISTER PCSRO
                                        ;       2-VERIFY INTERRUPT ENABLE BIT 06 = LOGIC 0
                                        ;       3-VERIFY DONE INTERRUPT BIT 11 = LOGIC 0
                                        ;       4-WRITE PORT COMMAND NOP TO PCSRO BITS 3:0
                                        ;       5-READ REGISTER PCSRO
                                        ;       6-VERIFY DONE INTERRUPT BIT 11 = LOGIC 1
                                        ;       7-WRITE REGISTER PCSRO BIT 11 WITH LOGIC 1
                                        ;       8-READ REGISTER PCSRO
                                        ;       9-VERIFY PCSRO BIT 11 = 0
                                        ;
                                        ;*****************************************************************

023740'                                 BGNTST
023740'                                                                         T8::
023740'                                  BGNSUB ;#1
023740'                                                                         T8.1:
023740' 104402                                                                          TRAP    C$BSUB
                                        ;
                                        ;CHECK THE INTERRUPT ENABLE BIT; IF SET CLEAR IT
                                        ;
023742' 013701  000336'                         MOV     PCSRO,R1                ;GET PCSRO ADDRESS
023746' 011102                                  MOV     (R1),R2                 ;GET CONTENTS
023750' 032702  000100                          BIT     #INTE,R2                ;IS INTERRUPT ENABLE SET?
023754' 001412                                  BEQ     10$                     ;NO
023756' 112711  000000                          MOVB    #0,(R1)                 ;CLEAR INTERRUPT ENABLE
023762' 011102                                  MOV     (R1),R2                 ;GET PCSRO CONTENTS
023764' 032702  000100                          BIT     #INTE,R2                ;IS INTERRUPT ENABLE CLEAR NOW?
023770' 001404                                  BEQ     10$                     ;YES
023772'                                         ERRHRD  022,NOPERR,RACMG4       ;NO,PRINT ERROR MESSAGE
023772' 104456                                                                          TRAP    C$ERHRD
023774' 000026                                                                          .WORD   22
023776' 001706'                                                                         .WORD   NOPERR
024000' 012642'                                                                         .WORD   RACMG4
024002'                                 10$:
024002'                                  ENDSUB ;#2
024002'                                                                         L10114:
024002' 104403                                                                          TRAP    C$ESUB
024004'                                  BGNSUB ;#2
024004'                                                                         T8.2:
024004' 104402                                                                          TRAP    C$BSUB
                                        ;
                                        ;CHECK THE DONE INTERRUPT BIT; IF SET WRITE ONE TO CLEAR IT
                                        ;
```

```
024006' 032702  004000               BIT     #DNI,R2                 ;IS DONE INTERRUPT CLEAR?
024012' 001413                       BEQ     20$                     ;YES
024014' 000302                       SWAB    R2                      ;NO,ORIENT UPPER & LOWER BYTES
024016' 110261  000001               MOVB    R2,1(R1)                ;SO WE CAN CLEAR UPPER BITS
024022' 011102                       MOV     (R1),R2                 ;GET PCSRO CONTENTS
024024' 032702  004000               BIT     #DNI,R2                 ;IS DNI CLEAR NOW?
024030' 001404                       BEQ     20$                     ;YES
024032'                              ERRHRD  023,NOPERR,RACMG7       ;NO,PRINT ERROR MESSAGE
024032' 104456                                                                       TRAP    C$ERHRD
024034' 000027                                                                       .WORD   23
024036' 001706'                                                                      .WORD   NOPERR
024040' 012670'                                                                      .WORD   RACMG7
024042'                      20$:
024042'                       ENDSUB ;#2
024042'                                                                      L10115:
024042' 104403                                                                       TRAP    C$ESUB
024044'                       BGNSUB ;#3
024044'                                                                      T8.3:
024044' 104402                                                                       TRAP    C$BSUB

                             ;
                             ;ISSUE NOP PORT COMMAND AND CHECK FOR DNI
                             ;
024046' 012777  000006  154262       MOV     #PNOP,@PCSRO            ;ISSUE NOP PORT COMMAND
024054' 012737  000176  000332'      MOV     #2*SECOND,METER         ;SETUP TIMER
024062' 004737  017316'              JSR     PC,CHKDNI               ;WAIT FOR DNI TO SET
024066' 103022                       BCC     30$                     ;OK
                                                                     ;ERROR DNI DID NOT SET!
                                                                     ;SETUP TO PRINT ERROR MESSAGE
024070' 012737  001000' 000310'      MOV     #$DNI,BITNAM            ;POINT TO 'DNI' ASCII STRING
024076' 012737  001277' 000312'      MOV     #$NSET,BITSTA           ;POINT TO 'NOT SET' ASCII STRING
024104' 012737  001342' 000314'      MOV     #$AFTER,PWHEN           ;POINT TO 'AFTER' ASCII STRING
024112' 012737  001405' 000316'      MOV     #$NOP,PCOMND            ;POINT TO 'NOP' ASCII STRING
024120'                              ERRHRD  024,NOPERR,MSG1         ;PRINT ERROR MESSAGE
024120' 104456                                                                       TRAP    C$ERHRD
024122' 000030                                                                       .WORD   24
024124' 001706'                                                                      .WORD   NOPERR
024126' 012716'                                                                      .WORD   MSG1
024130'                              ESCAPE  TST                     ;DON'T CONTINUE TEST IF ERROR OCCURRED
024130' 104410                                                                       TRAP    C$ESCAPE
024132' 000026                                                                       .WORD   L10113-.
024134'                      30$:
024134'                        BGNSEG
024134' 104404                                                                       TRAP    C$BSEG
                             ;
                             ;WRITE ONE TO CLEAR 'DNI'
                             ;
024136' 004737  017362'              JSR     PC,CLRDNI               ;GO CLEAR DNI BIT
024142' 103004                       BCC     40$                     ;IT CLEARED OK
                                                                     ;ERROR DNI DID NOT CLEAR!
024144'                              ERRHRD  025,NOPERR,RACMG7       ;NO,PRINT ERROR MESSAGE
024144' 104456                                                                       TRAP    C$ERHRD
024146' 000031                                                                       .WORD   25
024150' 001706'                                                                      .WORD   NOPERR
024152' 012670'                                                                      .WORD   RACMG7
024154'                      40$:
024154'                        ENDSEG
```

```
5774  024154'                                                    10000$:
5775  024154' 104405                                                     TRAP    C$ESEG
5776  024156'                  ENDSUB ;#3
5777  024156'                                                    L10116:
5778  024156' 104403                                                     TRAP    C$ESUB
5779  024160'                ENDTST
5780  024160'                                                    L10113:
5781  024160' 104401                                                     TRAP    C$ETST
5782
```

```
                                        .SBTTL  TEST 9: SELF TEST

                                        ;*****************************************************************
                                        ;
                                        ;       THIS TEST VERIFIES THAT THE ROM BASED SELF TEST
                                        ;       CAN BE RUN SUCCESSFULLY WHEN INVOKED VIA
                                        ;       THE SELF TEST PORT COMMAND.
                                        ;
                                        ;       TEST SEQUENCE:
                                        ;               1. CLEAR OUTSTANDING INTERRUPTS
                                        ;               2. ISSUE THE SELF TEST PORT COMMAND
                                        ;               3. WAIT FOR DNI
                                        ;               4. CHECK LITE BITE REGISTER FOR SUCCESSFUL SELF TEST
                                        ;               5. WRITE ONE TO CLEAR DNI
                                        ;
                                        ;*****************************************************************

 024162'                                BGNTST
 024162'                                                                        T9::
 024162' 004737  017410'                        JSR     PC,CLRINT               ;CLEAR ANY OUTSTANDING INTERRUPTS
 024166' 103006                                 BCC     10$                     ;OK
 024170'                                        ERRHRD  026,SLFTST              ;ERROR OCCURRED TRYING TO CLEAR
 024170' 104456                                                                         TRAP    C$ERHRD
 024172' 000032                                                                         .WORD   26
 024174' 001726'                                                                        .WORD   SLFTST
 024176' 000000                                                                         .WORD   0
 024200'                                        ESCAPE  TST                     ;LEAVE THIS TEST
 024200' 104410                                                                         TRAP    C$ESCAPE
 024202' 000202                                                                         .WORD   L10117-.
 024204' 012737  001364  000332' 10$:   MOV     #12.*SECOND,METER       ;PUT SOME TIME ON THE METER
 024212' 012777  000003  154116         MOV     #SLFT,@PCSR0            ; RUN SELF TEST
 024220' 004737  017316'                JSR     PC,CHKDNI               ; DNI ?
 024224' 103020                         BCC     30$                     ; YES
                                                                        ;ERROR DNI FAILED TO SET!
 024226' 012737  001000' 000310'        MOV     #SDNI,BITNAM
 024234' 012737  001277' 000312'        MOV     #SNSET,BITSTA
 024242' 012737  001342' 000314'        MOV     #SAFTER,PWHEN
 024250' 012737  001373' 000316'        MOV     #SSLFT,PCOMND
 024256'                                ERRHRD  027,SLFTST,MSG1         ;PRINT ERROR MESSAGE
 024256' 104456                                                                         TRAP    C$ERHRD
 024260' 000033                                                                         .WORD   27
 024262' 001726'                                                                        .WORD   SLFTST
 024264' 012716'                                                                        .WORD   MSG1
                                ;MAC001 ESCAPE  TST                     ; AND ABORT TEST
                                ;
 024266' 017703  154046         30$:    MOV     @PCSR1,R3               ;GET PCSR1 CONTENTS
 024272' 032703  037400                 BIT     #SFT,R3                 ;WAS SELF TEST SUCCESSFUL?
 024276' 001433                         BEQ     40$                     ;YES
 024300' 005103                         COM     R3                      ;GET TEST NUMBER CORRECT POLARITY
 024302' 042703  140377                 BIC     #^CSFT,R3               ;MASK ALL BUT SELF TEST FIELD
 024306' 012701  000010                 MOV     #8.,R1                  ;SHIFT RESULT OVER 8 BIT POSTIONS
 024312' 006203                 35$:    ASR     R3
 024314' 005301                         DEC     R1
 024316' 001375                         BNE     35$
 024320' 010304                         MOV     R3,R4                   ;SAVE ERROR NUMBER
 024322' 006303                         ASL     R3                      ;MAKE NUMBER A BYTE OFFSET
```

```
024324' 062703 024410'          ADD     #STTBL,R3               ; INDEX INTO SELF TEST TABLE
024330' 105773 000000           TSTB    @(R3)                   ;IS ERROR CODE UNUSED?          ;MAC001
024334' 001004                  BNE     36$                     ;NO, CODE IS OK                 ;MAC001
024336' 012737 027112' 024406'  MOV     #UNUSED,STMSG           ;BOGUS ERROR CODE MESSAGE       ;MAC001
024344' 000402                  BR      37$                                                     ;MAC001
024346' 011337 024406'    36$:  MOV     (R3),STMSG              ; LOAD INTO SELF TEST MESSAGE
024352'                   37$:  ERRHRD  028,SLFTST,MSG34        ; REPORT SELF TEST FAILURE
024352' 104456                                                          TRAP    C$ERHRD
024354' 000034                                                          .WORD   28
024356' 001726'                                                         .WORD   SLFTST
024360' 015622'                                                         .WORD   MSG34
024362'                         ESCAPE  TST                     ; AND ABORT TEST
024362' 104410                                                          TRAP    C$ESCAPE
024364' 000020                                                          .WORD   L10117-.
                          ;
024366' 004737 017362'    40$:  JSR     PC,CLRDNI               ; WRITE ONE TO CLEAR DNI
                                                                ; ERROR?
024372' 103004                  BCC     50$                     ; NO
                                                                ;ERROR DNI FAILED TO CLEAR!
024374'                         ERRHRD  029,SLFTST,RACMG7       ;PRINT ERROR MESSAGE
024374' 104456                                                          TRAP    C$ERHRD
024376' 000035                                                          .WORD   29
024400' 001726'                                                         .WORD   SLFTST
024402' 012670'                                                         .WORD   RACMG7
024404'                   50$:
024404'                         ENDTST
024404'                                                         L10117:
024404' 104401                                                          TRAP    C$ETST
```

```
                          ;
                          ;LOCAL STORAGE FOR TEST 9
                          ;
024406' 000000            STMSG::         .WORD   0               ; SELF TEST MESSAGE ADDRESS
                          ;
                          ;SELF TEST MESSAGE TABLE
                          ;
024410' 024610'           STTBL::         .WORD   SMSG00
024412' 024632'                           .WORD   SMSG01
024414' 024652'                           .WORD   SMSG02
024416' 024656'                           .WORD   SMSG03
024420' 024706'                           .WORD   SMSG04
024422' 024742'                           .WORD   SMSG05
024424' 024766'                           .WORD   SMSG06
024426' 025030'                           .WORD   SMSG07
024430' 025073'                           .WORD   SMSG10
024432' 025140'                           .WORD   SMSG11
024434' 025160'                           .WORD   SMSG12
024436' 025166'                           .WORD   SMSG13
024440' 025213'                           .WORD   SMSG14
024442' 025214'                           .WORD   SMSG15
024444' 025215'                           .WORD   SMSG16
024446' 025216'                           .WORD   SMSG17
024450' 025217'                           .WORD   SMSG20
024452' 025233'                           .WORD   SMSG21
024454' 025234'                           .WORD   SMSG22
024456' 025235'                           .WORD   SMSG23
024460' 025236'                           .WORD   SMSG24
024462' 025237'                           .WORD   SMSG25
024464' 025240'                           .WORD   SMSG26
024466' 025241'                           .WORD   SMSG27
024470' 025242'                           .WORD   SMSG30
024472' 025304'                           .WORD   SMSG31
024474' 025347'                           .WORD   SMSG32
024476' 025413'                           .WORD   SMSG33
024500' 025450'                           .WORD   SMSG34
024502' 025512'                           .WORD   SMSG35
024504' 025546'                           .WORD   SMSG36
024506' 025610'                           .WORD   SMSG37
024510' 025644'                           .WORD   SMSG40
024512' 025721'                           .WORD   SMSG41
024514' 025773'                           .WORD   SMSG42
024516' 026046'                           .WORD   SMSG43
024520' 026112'                           .WORD   SMSG44
024522' 026163'                           .WORD   SMSG45
024524' 026226'                           .WORD   SMSG46
024526' 026227'                           .WORD   SMSG47
024530' 026230'                           .WORD   SMSG50
024532' 026307'                           .WORD   SMSG51
024534' 026363'                           .WORD   SMSG52
024536' 026440'                           .WORD   SMSG53
024540' 026506'                           .WORD   SMSG54
024542' 026561'                           .WORD   SMSG55
024544' 026626'                           .WORD   SMSG56
024546' 026627'                           .WORD   SMSG57
024550' 026630'                           .WORD   SMSG60
```

```
024552' 026644'                                 .WORD   SMSG61
024554' 026670'                                 .WORD   SMSG62
024556' 026714'                                 .WORD   SMSG63
024560' 026734'                                 .WORD   SMSG64
024562' 026740'                                 .WORD   SMSG65
024564' 026752'                                 .WORD   SMSG66
024566' 026764'                                 .WORD   SMSG67
024570' 027000'                                 .WORD   SMSG70
024572' 027012'                                 .WORD   SMSG71
024574' 027036'                                 .WORD   SMSG72
024576' 027060'                                 .WORD   SMSG73
024600' 027061'                                 .WORD   SMSG74
024602' 027062'                                 .WORD   SMSG75
024604' 027063'                                 .WORD   SMSG76
024606' 027064'                                 .WORD   SMSG77
                                ;ASCII MESSAGES
024610' 042516  042526  020122  SMSG00::        .ASCIZ  /NEVER GOT STARTED/
024616' 047507  020124  052123
024624' 051101  042524  000104
024632' 050103  020125  047111  SMSG01::        .ASCIZ  /CPU INSTRUCTION/                       ;MAC001
024640' 052123  052522  052103
024646' 047511  000116
024652' 047522  000115          SMSG02::        .ASCIZ  /ROM/                                   ;MAC001
024656' 051127  052111  040505  SMSG03::        .ASCIZ  /WRITEABLE CONTROL STORE/               ;MAC001
024664' 046102  020105  047503
024672' 052116  047522  020114
024700' 052123  051117  000105
                                ;MAC001SMSG04:: .ASCIZ  /PHYSICAL ADDRESS ROM/
024706' 030524  020061  047125  SMSG04::        .ASCIZ  /T11 UNIBUS ADDRESS REGISTER/           ;MAC001
024714' 041111  051525  040440
024722' 042104  042522  051523
024730' 051040  043505  051511
024736' 042524  000122
024742' 042522  042503  053111  SMSG05::        .ASCIZ  /RECEIVER UNIBUS DMA/                   ;MAC001
024750' 051105  052440  044516
024756' 052502  020123  046504
024764' 000101
024766' 041520  051123  020061  SMSG06::        .ASCIZ  /PCSR1 LOWER BYTE AND T11 DMA READ/     ;MAC001
024774' 047514  042527  020122
025002' 054502  042524  040440
025010' 042116  052040  030461
025016' 042040  040515  051040
025024' 040505  000104
025030' 041520  051123  020060  SMSG07::        .ASCIZ  /PCSR0 UPPER BYTE AND T11 DMA WRITE/    ;MAC001
025036' 050125  042520  020122
025044' 054502  042524  040440
025052' 042116  052040  030461
025060' 042040  040515  053440
025066' 044522  042524     000
025073'    120  051503  030122  SMSG10::        .ASCIZ  /PCSR0 LOWER BYTE AND LINK MEMORY DMA/  ;MAC001
025100' 046040  053517  051105
025106' 041040  052131  020105
025114' 047101  020104  044514
025122' 045516  046440  046505
025130' 051117  020131  046504
025136' 000101
```

```
025140' 041520  051123  020062  SMSG11::        .ASCIZ  /PCSR2 AND PCSR3/                       ;MAC001
025146' 047101  020104  041520
025154' 051123  000063
025160' 044524  042515  000122  SMSG12::        .ASCIZ  /TIMER/                                 ;MAC001
025166' 044120  051531  041511  SMSG13::        .ASCIZ  /PHYSICAL ADDRESS ROM/                  ;MAC001
025174' 046101  040440  042104
025202' 042522  051523  051040
025210' 046517     000
025213'    000                  SMSG14::        .BYTE   0               ;UNUSED                 ;MAC001
025214'    000                  SMSG15::        .BYTE   0               ;UNUSED                 ;MAC001
025215'    000                  SMSG16::        .BYTE   0               ;UNUSED                 ;MAC001
025216'    000                  SMSG17::        .BYTE   0               ;UNUSED                 ;MAC001
025217'    114  047111  020113  SMSG20::        .ASCIZ  /LINK MEMORY/                           ;MAC001
025224' 042515  047515  054522
025232'    000
025233'    000                  SMSG21::        .BYTE   0               ;UNUSED                 ;MAC001
025234'    000                  SMSG22::        .BYTE   0               ;UNUSED                 ;MAC001
025235'    000                  SMSG23::        .BYTE   0               ;UNUSED                 ;MAC001
025236'    000                  SMSG24::        .BYTE   0               ;UNUSED                 ;MAC001
025237'    000                  SMSG25::        .BYTE   0               ;UNUSED                 ;MAC001
025240'    000                  SMSG26::        .BYTE   0               ;UNUSED                 ;MAC001
025241'    000                  SMSG27::        .BYTE   0               ;UNUSED                 ;MAC001
025242' 047514  040503  020114  SMSG30::        .ASCIZ  /LOCAL LOOPBACK -  XMITTER TIMEOUT/     ;MAC001
025250' 047514  050117  040502
025256' 045503  026440  020040
025264' 046530  052111  042524
025272' 020122  044524  042515
025300' 052517  000124
025304' 047514  040503  020114  SMSG31::        .ASCIZ  /LOCAL LOOPBACK -  RECEIVER TIMEOUT/    ;MAC001
025312' 047514  050117  040502
025320' 045503  026440  020040
025326' 042522  042503  053111
025334' 051105  052040  046511
025342' 047505  052125     000
025347'    114  041517  046101  SMSG32::        .ASCIZ  /LOCAL LOOPBACK -  BUFFER COMPARSION/   ;MAC001
025354' 046040  047517  041120
025362' 041501  020113  020055
025370' 041040  043125  042506
025376' 020122  047503  050115
025404' 051101  044523  047117
025412'    000
025413'    114  041517  046101  SMSG33::        .ASCIZ  /LOCAL LOOPBACK -  BYTE COUNT/          ;MAC001
025420' 046040  047517  041120
025426' 041501  020113  020055
025434' 041040  052131  020105
025442' 047503  047125  000124
025450' 047514  040503  020114  SMSG34::        .ASCIZ  /LOCAL LOOPBACK -  RECEIVER STATUS/     ;MAC001
025456' 047514  050117  040502
025464' 045503  026440  020040
025472' 042522  042503  053111
025500' 051105  051440  040524
025506' 052524  000123
025512' 047514  040503  020114  SMSG35::        .ASCIZ  /LOCAL LOOPBACK -  CRC ERROR/           ;MAC001
025520' 047514  050117  040502
025526' 045503  026440  020040
025534' 051103  020103  051105
```

```
6035  025542' 047522  000122
6036  025546' 047514  040503  020114  SMSG36::       .ASCIZ  /LOCAL LOOPBACK -  MATCH BIT ERROR/            ;MAC001
6037  025554' 047514  050117  040502
6038  025562' 045503  026440  020040
6039  025570' 040515  041524  020110
6040  025576' 044502  020124  051105
6041  025604' 047522  000122
6042  025610' 047514  040503  020114  SMSG37::       .ASCIZ  /LOCAL LOOPBACK -  TDR ERROR/                  ;MAC001
6043  025616' 047514  050117  040502
6044  025624' 045503  026440  020040
6045  025632' 042124  020122  051105
6046  025640' 047522  000122
6047  025644' 046530  052111  042524  SMSG40::       .ASCIZ  /XMITTER BUFFER ADDRESS - TRANSMITTER TIMEOUT/  ;MAC001
6048  025652' 020122  052502  043106
6049  025660' 051105  040440  042104
6050  025666' 042522  051523  026440
6051  025674' 052040  040522  051516
6052  025702' 044515  052124  051105
6053  025710' 052040  046511  047505
6054  025716' 052125     000
6055  025721'    130  044515  052124  SMSG41::       .ASCIZ  /XMITTER BUFFER ADDRESS - RECEIVER TIMEOUT/     ;MAC001
6056  025726' 051105  041040  043125
6057  025734' 042506  020122  042101
6058  025742' 051104  051505  020123
6059  025750' 020055  042522  042503
6060  025756' 053111  051105  052040
6061  025764' 046511  047505  052125
6062  025772'    000
6063  025773'    130  044515  052124  SMSG42::       .ASCIZ  /XMITTER BUFFER ADDRESS - BUFFER COMPARSION/    ;MAC001
6064  026000' 051105  041040  043125
6065  026006' 042506  020122  042101
6066  026014' 051104  051505  020123
6067  026022' 020055  052502  043106
6068  026030' 051105  041440  046517
6069  026036' 040520  051522  047511
6070  026044' 000116
6071  026046' 046530  052111  042524  SMSG43::       .ASCIZ  /XMITTER BUFFER ADDRESS - BYTE COUNT/           ;MAC001
6072  026054' 020122  052502  043106
6073  026062' 051105  040440  042104
6074  026070' 042522  051523  026440
6075  026076' 041040  052131  020105
6076  026104' 047503  047125  000124
6077  026112' 046530  052111  042524  SMSG44::       .ASCIZ  /XMITTER BUFFER ADDRESS - RECEIVER STATUS/      ;MAC001
6078  026120' 020122  052502  043106
6079  026126' 051105  040440  042104
6080  026134' 042522  051523  026440
6081  026142' 051040  041505  044505
6082  026150' 042526  020122  052123
6083  026156' 052101  051525     000
6084  026163'    130  044515  052124  SMSG45::       .ASCIZ  /XMITTER BUFFER ADDRESS - CRC ERROR/            ;MAC001
6085  026170' 051105  041040  043125
6086  026176' 042506  020122  042101
6087  026204' 051104  051505  020123
6088  026212' 020055  051103  020103
6089  026220' 051105  047522  000122
6090  026226'    000                  SMSG46::       .BYTE  0            ;UNUSED                            ;MAC001
```

```
026227'    000                   SMSG47::        .BYTE   0                   ;UNUSED                             ;MAC001
026230' 042522  042503  053111   SMSG50::        .ASCIZ  /RECEIVER BUFFER ADDRESS -  TRANSMITTER TIMEOUT/;MAC001
026236' 051105  041040  043125
026244' 042506  020122  042101
026252' 051104  051505  020123
026260' 020055  052040  040522
026266' 051516  044515  052124
026274' 051105  052040  046511
026302' 047505  052125     000
026307'    122  041505  044505   SMSG51::        .ASCIZ  /RECEIVER BUFFER ADDRESS -  RECEIVER TIMEOUT/    ;MAC001
026314' 042526  020122  052502
026322' 043106  051105  040440
026330' 042104  042522  051523
026336' 026440  020040  042522
026344' 042503  053111  051105
026352' 052040  046511  047505
026360' 052125     000
026363'    122  041505  044505   SMSG52::        .ASCIZ  /RECEIVER BUFFER ADDRESS -  BUFFER COMPARSION/  ;MAC001
026370' 042526  020122  052502
026376' 043106  051105  040440
026404' 042104  042522  051523
026412' 026440  020040  052502
026420' 043106  051105  041440
026426' 046517  040520  051522
026434' 047511  000116
026440' 042522  042503  053111   SMSG53::        .ASCIZ  /RECEIVER BUFFER ADDRESS -  BYTE COUNT/          ;MAC001
026446' 051105  041040  043125
026454' 042506  020122  042101
026462' 051104  051505  020123
026470' 020055  041040  052131
026476' 020105  047503  047125
026504' 000124
026506' 042522  042503  053111   SMSG54::        .ASCIZ  /RECEIVER BUFFER ADDRESS -  RECEIVER STATUS/     ;MAC001
026514' 051105  041040  043125
026522' 042506  020122  042101
026530' 051104  051505  020123
026536' 020055  051040  041505
026544' 044505  042526  020122
026552' 052123  052101  051525
026560'    000
026561'    122  041505  044505   SMSG55::        .ASCIZ  /RECEIVER BUFFER ADDRESS -  CRC ERROR/           ;MAC001
026566' 042526  020122  052502
026574' 043106  051105  040440
026602' 042104  042522  051523
026610' 026440  020040  051103
026616' 020103  051105  047522
026624' 000122
026626'    000                   SMSG56::        .BYTE   0                   ;UNUSED                             ;MAC001
026627'    000                   SMSG57::        .BYTE   0                   ;UNUSED                             ;MAC001
026630' 052522  052116  050040   SMSG60::        .ASCIZ  /RUNT PACKET/                                           ;MAC001
026636' 041501  042513  000124
026644' 044515  044516  052515   SMSG61::        .ASCIZ  /MINIMUM PACKET SIZE/                                   ;MAC001
026652' 020115  040520  045503
026660' 052105  051440  055111
026666' 000105
026670' 040515  044530  052515   SMSG62::        .ASCIZ  /MAXIMUM PACKET SIZE/                                   ;MAC001
```

```
026676' 020115  040520  045503
026704' 052105  051440  055111
026712' 000105
026714' 053117  051105  044523  SMSG63::        .ASCIZ  /OVERSIZE PACKET/                       ;MAC001
026722' 042532  050040  041501
026730' 042513  000124
026734' 051103  000103          SMSG64::        .ASCIZ  /CRC/                                   ;MAC001
026740' 047503  046114  051511  SMSG65::        .ASCIZ  /COLLISION/                             ;MAC001
026746' 047511  000116
026752' 042510  051101  041124  SMSG66::        .ASCIZ  /HEARTBEAT/                             ;MAC001
026760' 040505  000124
026764' 040510  043114  042040  SMSG67::        .ASCIZ  /HALF DUPLEX/                           ;MAC001
026772' 050125  042514  000130
027000' 052515  052114  041511  SMSG70::        .ASCIZ  /MULTICAST/                             ;MAC001
027006' 051501  000124
027012' 042101  051104  051505  SMSG71::        .ASCIZ  /ADDRESS RECOGNITION/                   ;MAC001
027020' 020123  042522  047503
027026' 047107  052111  047511
027034' 000116
027036' 054105  042524  047122  SMSG72::        .ASCIZ  /EXTERNAL LOOPBACK/                     ;MAC001
027044' 046101  046040  047517
027052' 041120  041501  000113
027060'    000                  SMSG73::        .BYTE   0               ;UNUSED                 ;MAC001
027061'    000                  SMSG74::        .BYTE   0               ;UNUSED                 ;MAC001
027062'    000                  SMSG75::        .BYTE   0               ;UNUSED                 ;MAC001
027063'    000                  SMSG76::        .BYTE   0               ;UNUSED                 ;MAC001
027064' 047503  050115  042514  SMSG77::        .ASCIZ  /COMPLETED - NO ERRORS/
027072' 042524  020104  020055
027100' 047516  042440  051122
027106' 051117  000123
027112' 047125  042504  044506  UNUSED::        .ASCIZ  /UNDEFINED/                             ;MAC001
027120' 042516  000104
                                                .EVEN
```

```
                                      .SBTTL  TEST 10: DEUNA ROM DUMP TEST

                                      ;*************************************************************************
                                      ;
                                      ;THIS TEST WILL VERIFY THAT THE DATA PATH FROM THE T11 PROCESSOR
                                      ;TO THE UNIBUS INTERFACE IS INTACT AND ABLE TO TRANSFER DATA RELIABLY.
                                      ;THIS DATA PATH IS CRUCIAL FOR FURTHER TESTING BECAUSE IT IS NECESSARY
                                      ;FOR LOADING REPAIR-LEVEL DIAGNOSTICS INTO THE WCS.
                                      ;
                                      ;THE TEST STRATEGY IS TO TRANSFER KNOWN DATA OVER THE DATA PATH AND TO
                                      ;VERIFY THE TRANSFERRED DATA.
                                      ;
                                      ;THE DATA SOURCE FOR THE DUMP TEST IS THE ROM MICROCODE RESIDENT ON THE
                                      ;DEUNA PORT BOARD. A DUMP OF THE ROM WILL EXERCISE THE DATA PATH NEEDED
                                      ;FOR LOADING WCS AND THE ROM CONTENTS CAN BE VERIFIED. THE ROM MICROCODE
                                      ;WILL BE CHECKED BY VERIFYING THE CRC BYTES. THE CRC BYTES CHARACTERIZE
                                      ;THE DATA CONTENTS OF THE ROM AND ARE BURNED INTO THE ROM AT THE TIME OF
                                      ;MANUFACTURE. A FAILURE TO VERIFY THE CRC CALCULATION ON THE DUMPED ROM
                                      ;DATA DUMP WILL BE INTERPRETED AS AN ERROR IN THE DATA PATH.
                                      ;
                                      ;TEST SEQUENCE:
                                      ;
                                      ;       1-WRITE PCSR2 AND PCSR3 WITH THE ADDRESS OF THE PORT CONTROL BLOCK
                                      ;       2-WRITE <GET PCBB> PORT COMMAND TO PCSR0
                                      ;       3-READ PCSR0 AND VERIFY DNI BIT SET
                                      ;       4-WRITE PCSR0 DNI BIT TO RESET IT
                                      ;       5-FILL MEMORY BUFFER WITH A BACKGROUND PATTERN
                                      ;       6-WRITE PORT CONTROL BLOCK WITH 'DUMP INTERNAL MEMORY' FUNCTION CODE
                                      ;       7-WRITE PORT CONTROL BLOCK WITH UNIBUS DATA BLOCK BASE ADDRESS
                                      ;       8-WRITE DATA BLOCK LENGTH TO UNIBUS DATA BLOCK
                                      ;       9-WRITE MEMORY BUFFER ADDRESS TO UNIBUS DATA BLOCK
                                      ;       10-WRITE INTERNAL DATA BLOCK ADDRESS TO UNIBUS DATA BLOCK
                                      ;       11-WRITE <GET CMD> PORT COMMAND TO PCSR0
                                      ;       12-READ PCSR0 AND VERIFY DNI SET
                                      ;       13-CALCULATE CRC ON DUMPED DATA
                                      ;       14-REPEAT STEPS 5-13 ON EACH 1K OF ROM
                                      ;       15-VERIFY CRC =0
                                      ;
                                      ;*************************************************************************

        027124'                       BGNTST
        027124'                                                                   T10::
        027124'                        BGNSUB ;#1
        027124'                                                                   T10.1:
        027124'  104402                                                                   TRAP    C$BSUB
                                      ;
                                      ;LOAD PCSR2+3; ISSUE GET PORT CONTROL BLOCK PORT COMMAND; AND WAIT FOR 'DNI'
                                      ;
        027126'                         BGNSEG
        027126'  104404                                                                   TRAP    C$BSEG
        027130'  012777  000606' 151204         MOV     #PCBB,@PCSR2          ;LOAD PCSR2+3 WITH PORT CONTROL BLOCK..
        027136'  005077  151202                 CLR     @PCSR3                ;ADDRESS
        027142'  012777  000001  151166         MOV     #GETPCB,@PCSR0        ;ISSUE GET PORT CONTROL BLOCK COMMAND
        027150'  012737  000176  000332'        MOV     #2*SECOND,METER ;PUT SOME TIME IN THE METER
        027156'  004737  017316'                JSR     PC,CHKDNI             ;WAIT FOR DNI TO SET
        027162'  103022                         BCC     20$                   ;OK
```

```
                                                                 ;ERROR DNI DID NOT SET!
027164' 012737 001000' 000310'      MOV     #$DNI,BITNAM         ;NO, FORMAT ERROR MESSAGE
027172' 012737 001277' 000312'      MOV     #$NSET,BITSTA
027200' 012737 001342' 000314'      MOV     #$AFTER,PWHEN
027206' 012737 001350' 000316'      MOV     #$G:PCB,PCOMND
027214'                             ERRHRD  030,ROMDMP,MSG1 ;PRINT ERROR MESSAGE
027214' 104456                                                                 TRAP    C$ERHRD
027216' 000036                                                                 .WORD   30
027220' 001747'                                                                .WORD   ROMDMP
027222' 012716'                                                                .WORD   MSG1
027224'                             ESCAPE  TST                  ;CAN NOT CONTINUE TESTING
027224' 104410                                                                 TRAP    C$ESCAPE
027226' 000310                                                                 .WORD   L10120-.
027230'                       20$:
027230'                         ENDSEG
027230'                                                             10000$:
027230' 104405                                                                 TRAP    C$ESEG
027232'                         BGNSEG
027232' 104404                                                                 TRAP    C$BSEG
                              ;
                              ;WRITE ONE TO CLEAR 'DNI'
                              ;
027234' 004737 017362'              JSR     PC,CLRDNI            ;GO CLEAR DNI
027240' 103004                      BCC     25$                  ;OK
                                                                 ;ERROR DNI FAILED TO CLEAR!
027242'                             ERRHRD  031,ROMDMP,RACMG7    ;PRINT ERROR MESSAGE
027242' 104456                                                                 TRAP    C$ERHRD
027244' 000037                                                                 .WORD   31
027246' 001747'                                                                .WORD   ROMDMP
027250' 012670'                                                                .WORD   RACMG7
027252'                       25$:
027252'                         ENDSEG
027252'                                                             10001$:
027252' 104405                                                                 TRAP    C$ESEG
027254'                        ENDSUB ;#1
027254'                                                             L10121:
027254' 104403                                                                 TRAP    C$ESUB
027256'                        BGNSUB ;#2
027256'                                                             T10.2:
027256' 104402                                                                 TRAP    C$BSUB
                              ;
                              ;FILL BUFFER WITH PATTERN; FORMAT PCBB AND UDBB; ISSUE GET COMMAND PORT COMMAND
                              ;AND WAIT FOR 'DNI'
                              ;
027260' 012737 000020  000606'      MOV     #DIM,PCBB            ;LOAD PCB WITH DUMP INTERNAL MEMORY FUNCTION
027266' 012737 000616' 000610'      MOV     #UDBB,PCBB+2         ;LOAD UNIBUS DATA BLOCK BASE ADDRESS...
027274' 005037 000612'              CLR     PCBB+4               ;INTO PCB
027300' 012737 040000  000624'      MOV     #ROMADR,UDBB+6       ;INTERNAL BASE ADDRESS OF ROM
027306' 012705 000010               MOV     #8.,R5               ;DUMP ROM IN 8. 1K CHUNKS
027312' 005004                      CLR     R4                   ;INITIAL CRC VALUE
027314'                       26$:
027314'                         BGNSEG
027314' 104404                                                                 TRAP    C$BSEG
027316' 013701 000324'              MOV     FREMEM,R1            ;GET POINTER TO BUFFER
027322' 012702 002000               MOV     #SIZ1K/2,R2          ;GET 1K WORD COUNT
027326' 012703 000520'              MOV     #PAT1,R3             ;GET A BACKGROUND PATTERN
```

```
027332' 011321                 27$:    MOV     (R3),(R1)+              ;FILL BUFFER WITH BACKGROUND PATTERN
027334' 005302                         DEC     R2
027336' 001375                         BNE     27$
027340' 012737  004000  000616'        MOV     #ROMSIZ/8.,UDBB         ;1K BYTE COUNT
027346' 013737  000324' 000620'        MOV     FREMEM,UDBB+2           ;AND BUFFER ADDRESS
027354' 005037  000622'                CLR     UDBB+4
027360' 012737  000176  000332'        MOV     #2*SECOND,METER         ;PUT SOME TIME ON THE METER
027366' 012777  000002  150742         MOV     #GETCMD,@PCSR0          ;LOAD 'GET COMMAND' PORT COMMAND
027374' 004737  017316'                JSR     PC,CHKDNI               ;WAIT FOR DNI TO SET
027400' 103022                         BCC     30$                     ;OK
                                                                       ;ERROR, DNI DID NOT SET!
027402' 012737  001000' 000310'        MOV     #$DNI,BITNAM            ;FORMAT ERROR MESSAGE
027410' 012737  001277' 000312'        MOV     #$NSET,BITSTA
027416' 012737  001342' 000314'        MOV     #$AFTER,PWHEN
027424' 012737  001357' 000316'        MOV     #$GTCMD,PCOMND
027432'                                ERRHRD  032,ROMDMP,MSG1
027432' 104456                                                                  TRAP    C$ERHRD
027434' 000040                                                                  .WORD   32
027436' 001747'                                                                 .WORD   ROMDMP
027440' 012716'                                                                 .WORD   MSG1
027442'                                ESCAPE  TST
027442' 104410                                                                  TRAP    C$ESCAPE
027444' 000072                                                                  .WORD   L10120-.
027446'                        30$:
027446'                          ENDSEG
027446'                                                              10000$:
027446' 104405                                                                  TRAP    C$ESEG
027450'                          BGNSEG
027450' 104404                                                                  TRAP    C$BSEG
                               ;
                               ;WRITE ONE TO CLEAR 'DNI'
                               ;
027452' 004737  017362'                JSR     PC,CLRDNI               ;GO CLEAR DNI BIT
027456' 103004                         BCC     33$                     ;OK
                                                                       ;ERROR DNI FAILED TO CLEAR
027460'                                ERRHRD  033,ROMDMP,RACMG7       ;PRINT ERROR MESSAGE
027460' 104456                                                                  TRAP    C$ERHRD
027462' 000041                                                                  .WORD   33
027464' 001747'                                                                 .WORD   ROMDMP
027466' 012670'                                                                 .WORD   RACMG7
027470'                        33$:
027470'                          ENDSEG
027470'                                                              10001$:
027470' 104405                                                                  TRAP    C$ESEG
                               ;
                               ;CALCULATE CRC ON 1K OF DATA DUMPED FROM ROM
                               ;
027472' 013700  000324'                MOV     FREMEM,R0               ;GET BUFFER ADDRESS
027476' 012702  004000                 MOV     #SIZ1K,R2               ;GET BYTE COUNT
027502' 004737  017030'                JSR     PC,CRC16                ;CALC CRC ON 1K BUFFER
027506' 062737  004000  000624'        ADD     #ROMSIZ/8.,UDBB+6       ;POINT TO NEXT 1K OF ROM
027514' 005305                         DEC     R5                      ;HAVE WE DONE ALL 8K?
027516' 001276                         BNE     26$                     ;NO
027520' 005704                         TST     R4                      ;YES, IS CRC = 0?
027522' 001404                         BEQ     40$                     ;YES, OK
027524'                                ERRHRD  034,ROMDMP,MSG2         ;PRINT ERROR MESSAGE
```

```
027524' 104456                                                      TRAP    C$ERHRD
027526' 000042                                                      .WORD   34
027530' 001747'                                                     .WORD   ROMDMP
027532' 012764'                                                     .WORD   MSG2
027534'                         40$:
027534'                          ENDSUB ;#2
027534'                                                     L10122:
027534' 104403                                                      TRAP    C$ESUB
027536'                         ENDTST
027536'                                                     L10120:
027536' 104401                                                      TRAP    C$ETST
```

```

                                        .SBTTL  TEST 11: WCS LOAD/DUMP TEST

                                        ;*************************************************************************
                                        ;
                                        ;THIS TEST WILL USE THE LOAD/DUMP PORT COMMAND TO VERIFY THE DATA
                                        ;PATHWAY TO/FROM THE WCS. PATTERNS WILL BE USED TO CHECK THE DATA PATHWAY
                                        ;FOR ALL SA0 AND SA1 ERRORS.
                                        ;
                                        ;BECAUSE THE OPERATIONAL MICROCODE NEEDS THE LOWER 2K OF WCS ONLY THE TOP HALF
                                        ; OF WCS WILL BE LOADED WITH A DATA PATTERN THEN DUMPED BACK
                                        ;TO MEMORY FOR VERIFICATION. THIS PROCEDURE WILL BE REPEATED FOR ALL PATTERNS.
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-FORMAT UNIBUS DATA BLOCK WITH NUMBER OF WORDS, WCS DESTINATION
                                        ;         ADDRESS, AND SOURCE BUFFER ADDRESS.
                                        ;       2-FILL SOURCE BUFFER WITH DATA PATTERN
                                        ;       3-WRITE PORT CONTROL BLOCK WITH ADDRESS OF UNIBUS DATA BLOCK
                                        ;         AND WITH THE 'LOAD INTERNAL MEMORY' FUNCTION CODE
                                        ;       4-WRITE PCSR2 AND PCSR3 WITH PORT CONTROL BLOCK ADDRESS
                                        ;       5-WRITE PCSR0 WITH <GET CMD> PORT COMMAND
                                        ;       6-READ PCSR0 TO VERIFY DNI SET
                                        ;       7-WRITE ONE TO CLEAR DNI
                                        ;       8-FILL DESTINATION BUFFER WITH ZEROS
                                        ;       9-WRITE PORT CONTROL BLOCK WITH 'DUMP INTERNAL MEMORY'
                                        ;         FUNCTION CODE.
                                        ;       10-WRITE PCSR0 WITH <GET CMD> PORT COMMAND
                                        ;       11-VERIFY DNI SET
                                        ;       12-WRITE ONE TO CLEAR DNI
                                        ;       13-COMPARE DESTINATION BUFFER WITH DATA PATTERN
                                        ;       14-REPEAT STEPS 1-13 FOR ALL DATA PATTERNS
                                        ;
                                        ;*************************************************************************

 027540'                                BGNTST
 027540'                                                                        T11::
 027540' 012704  000520'                        MOV     #PATERN,R4              ;POINT TO LIST OF DATA PATTERNS
 027544'                                 BGNSUB ;#1
 027544'                                                                        T11.1:
 027544' 104402                                                                         TRAP    C$BSUB
                                        ;
                                        ;LOAD WCS
                                        ;
 027546'                                 BGNSEG
 027546' 104404                                                                         TRAP    C$BSEG
                                        ;
                                        ;FILL SOURCE BUFFER; FORMAT UDBB AND PCBB; ISSUE GET PORT CONTROL BLOCK PORT
                                        ;COMMAND AND WAIT FOR 'DNI'
                                        ;
 027550' 013702  000324'                        MOV     FREMEM,R2               ;GET UNIBUS BUFFER ADDRESS
 027554' 012703  004000                         MOV     #WCSSIZ/4,R3            ;GET HALF SIZE OF WCS IN WORDS
 027560' 011422                         10$:    MOV     (R4),(R2)+              ;FILL BUFFER WITH DATA PATTERN
 027562' 005303                                 DEC     R3
 027564' 001375                                 BNE     10$
                                                                                ;SETUP UNIBUS DATA BLOCK
```

```
027566' 012737  010000  000616'          MOV     #WCSSIZ/2,UDBB            ;BYTE COUNT
027574' 013737  000324' 000620'          MOV     FREMEM,UDBB+2             ;BUFFER ADDRESS
027602' 005037  000622'                  CLR     UDBB+4
027606' 012737  010000  000624'          MOV     #WCSADR+<WCSSIZ/2>,UDBB+6 ;BASE ADDRESS OF TOP HALF OF WCS
                                                                           ;SETUP PORT CONTROL BLOCK
027614' 012737  000021  000606'          MOV     #LIM,PCBB                 ;'LOAD INTERNAL MEMORY' FUNCTION
027622' 012737  000616' 000610'          MOV     #UDBB,PCBB+2              ;ADDRESS OF UNIBUS DATA BLOCK
027630' 005037  000612'                  CLR     PCBB+4
027634' 012777  000606' 150500           MOV     #PCBB,@PCSR2              ;LOAD PCSR2+3 WITH PCB ADDRESS
027642' 005077  150476                   CLR     @PCSR3

027646' 012777  000001  150462           MOV     #GETPCB,@PCSR0            ;ISSUE 'GET PORT CONTROL BLOCK' PORT COMMAND
027654' 012737  000176  000332'          MOV     #2*SECOND,METER           ;PUT SOME TIME ON THE METER
027662' 004737  017316'                  JSR     PC,CHKDNI                 ;WAIT FOR DNI TO SET
027666' 103022                           BCC     20$                       ;OK
                                                                           ;ERROR DNI DID NOT SET!
027670' 012737  001000' 000310'          MOV     #$DNI,BITNAM              ;POINT TO 'DNI' ASCII STRING
027676' 012737  001277' 000312'          MOV     #$NSET,BITSTA             ;POINT TO 'NOT SET' ASCII STRING
027704' 012737  001342' 000314'          MOV     #$AFTER,PWHEN             ;POINT TO 'AFTER' ASCII STRING
027712' 012737  001350' 000316'          MOV     #$GTPCB,PCOMND            ;POINT TO 'GET PORT CONTROL BLOCK' ASCII STRING
027720'                                  ERRHRD  035,DATALD,MSG1           ;PRINT ERROR MESSAGE
027720' 104456                                                                                     TRAP    C$ERHRD
027722' 000043                                                                                     .WORD   35
027724' 001774'                                                                                    .WORD   DATALD
027726' 012716'                                                                                    .WORD   MSG1
027730'                                  ESCAPE  TST
027730' 104410                                                                                     TRAP    C$ESCAPE
027732' 000372                                                                                     .WORD   L10123-.
027734'                          20$:
027734'                            ENDSEG
027734'                                                                                  10000$:
027734' 104405                                                                                     TRAP    C$ESEG
027736'                            BGNSEG
027736' 104404                                                                                     TRAP    C$BSEG
                                 ;
                                 ;WRITE ONE TO CLEAR 'DNI'
                                 ;
027740' 004737  017362'                  JSR     PC,CLRDNI                 ;GO CLEAR DNI
027744' 103004                           BCC     25$                       ;OK
                                                                           ;ERROR DNI FAILED TO CLEAR
027746'                                  ERRHRD  036,DATALD,RACMG7         ;PRINT ERROR MESSAGE
027746' 104456                                                                                     TRAP    C$ERHRD
027750' 000044                                                                                     .WORD   36
027752' 001774'                                                                                    .WORD   DATALD
027754' 012670'                                                                                    .WORD   RACMG7
027756'                          25$:
027756'                            ENDSEG
027756'                                                                                  10001$:
027756' 104405                                                                                     TRAP    C$ESEG
027760'                            BGNSEG
027760' 104404                                                                                     TRAP    C$BSEG
                                 ;
                                 ;ISSUE GET COMMAND PORT COMMAND AND WAIT FOR 'DNI'
                                 ;
027762' 012777  000002  150346           MOV     #GETCMD,@PCSR0            ;ISSUE GET COMMAND PORT COMMAND
027770' 012737  000275  000332'          MOV     #3*SECOND,METER           ;PUT SOME TIME ON THE METER
```

```
027776' 004737  017316'             JSR     PC,CHKDNI               ;GO WAIT FOR DNI
030002' 103022                      BCC     30$                     ;OK
                                                                    ;ERROR DNI DID NOT SET!
030004' 012737  001000' 000310'     MOV     #$DNI,BITNAM
030012' 012737  001277' 000312'     MOV     #$NSET,BITSTA
030020' 012737  001342' 000314'     MOV     #$AFTER,PWHEN
030026' 012737  001357' 000316'     MOV     #$GTCMD,PCOMND
030034'                             ERRHRD  037,DATALD,MSG1         ;PRINT ERROR MESSAGE
030034' 104456                                                                  TRAP    C$ERHRD
030036' 000045                                                                  .WORD   37
030040' 001774'                                                                 .WORD   DATALD
030042' 012716'                                                                 .WORD   MSG1
030044'                             ESCAPE TST
030044' 104410                                                                  TRAP    C$ESCAPE
030046' 000256                                                                  .WORD   L10123-.
030050'                     30$:
030050'                       ENDSEG
030050'                                                             10002$:
030050' 104405                                                                  TRAP    C$ESEG
030052'                       BGNSEG
030052' 104404                                                                  TRAP    C$BSEG
                            ;
                            ;WRITE ONE TO CLEAR 'DNI'
                            ;
030054' 004737  017362'             JSR     PC,CLRDNI               ;GO CLEAR DNI
030060' 103004                      BCC     35$                     ;OK
                                                                    ;ERROR DNI FAILED TO CLEAR
030062'                             ERRHRD  038,DATALD,RACMG7       ;PRINT ERROR MESSAGE
030062' 104456                                                                  TRAP    C$ERHRD
030064' 000046                                                                  .WORD   38
030066' 001774'                                                                 .WORD   DATALD
030070' 012670'                                                                 .WORD   RACMG7
030072'                     35$:
030072'                       ENDSEG
030072'                                                             10003$:
030072' 104405                                                                  TRAP    C$ESEG
030074'                      ENDSUB ;#1
030074'                                                             L10124:
030074' 104403                                                                  TRAP    C$ESUB
030076'                      BGNSUB ;#2
030076'                                                             T11.2:
030076' 104402                                                                  TRAP    C$BSUB
                            ;
                            ;DUMP WCS
                            ;
030100'                       BGNSEG
030100' 104404                                                                  TRAP    C$BSEG
                            ;
                            ;CLEAR DESTINATION BUFFER; FORMAT PCBB AND UDBB; ISSUE GET COMMAND PORT COMMAND
                            ;AND WAIT FOR 'DNI'
                            ;
030102' 013702  000324'             MOV     FREMEM,R2               ;GET ADDRESS OF FREE MEMORY
030106' 012703  004000              MOV     #WCSSIZ/4,R3            ;SIZE IN WORDS OF BUFFER
030112' 005022              40$:    CLR     (R2)+                   ;FILL DESTINATION BUFFER WITH ZEROS
030114' 005303                      DEC     R3
030116' 001375                      BNE     40$
```

```
030120' 013737  000324' 000620'         MOV     FREMEM,UDBB+2
030126' 005037  000622'                 CLR     UDBB+4
030132' 012737  000020  000606'         MOV     #DIM,PCBB               ;LOAD 'DUMP INTERNAL MEMORY' FUNCTION
030140' 012777  000002  150170          MOV     #GETCMD,@PCSR0          ;ISSUE GET COMMAND PORT COMMAND
030146' 012737  000275  000332'         MOV     #3*SECOND,METER         ;PUT SOME TIME ON THE METER
030154' 004737  017316'                 JSR     PC,CHKDNI               ;WAIT FOR DNI TO SET
030160' 103022                          BCC     50$                     ;OK
030162' 012737  001000' 000310'         MOV     #$DNI,BITNAM
030170' 012737  001277' 000312'         MOV     #$NSET,BITSTA
030176' 012737  001342' 000314'         MOV     #$AFTER,PWHEN
030204' 012737  001357' 000316'         MOV     #$GTCMD,PCOMND
030212'                                 ERRHRD  039,DATALD,MSG1
030212' 104456                                                                  TRAP    C$ERHRD
030214' 000047                                                                  .WORD   39
030216' 001774'                                                                 .WORD   DATALD
030220' 012716'                                                                 .WORD   MSG1
030222'                                 ESCAPE TST
030222' 104410                                                                  TRAP    C$ESCAPE
030224' 000100                                                                  .WORD   L10123-.
030226'                         50$:
030226'                           ENDSEG
030226'                                                                 10000$:
030226' 104405                                                                  TRAP    C$ESEG
030230'                           BGNSEG
030230' 104404                                                                  TRAP    C$BSEG
030232' 004737  017362'                 JSR     PC,CLRDNI               ;GO CLEAR DNI
030236' 103004                          BCC     60$                     ;OK
                                                                        ;ERROR DNI FAILED TO CLEAR
030240'                                 ERRHRD  040,DATALD,RACMG7       ;PRINT ERROR MESSAGE
030240' 104456                                                                  TRAP    C$ERHRD
030242' 000050                                                                  .WORD   40
030244' 001774'                                                                 .WORD   DATALD
030246' 012670'                                                                 .WORD   RACMG7
030250'                         60$:
030250'                           ENDSEG
030250'                                                                 10001$:
030250' 104405                                                                  TRAP    C$ESEG

                                ;
                                ;COMPARE DUMPED DATA TO WRITTEN PATTERN
                                ;
030252' 013701  000324'                 MOV     FREMEM,R1               ;SOURCE BUFFER ADDRESS
030256' 012703  004000                  MOV     #<WCSSIZ/2>/2,R3        ;# OF WORDS TO COMPARE
030262' 021114                  70$:    CMP     (R1),(R4)               ;IS WHAT WAS LOADED SAME AS...
                                                                        ;WHAT WAS DUMPED?
030264' 001404                          BEQ     80$                     ;YES
030266'                                 ERRHRD  041,DATALD,MSG5 ;ERROR DATA COMPARE
030266' 104456                                                                  TRAP    C$ERHRD
030270' 000051                                                                  .WORD   41
030272' 001774'                                                                 .WORD   DATALD
030274' 013124'                                                                 .WORD   MSG5
030276' 005721                  80$:    TST     (R1)+                   ;POINT TO NEXT LOCATION
030300' 005303                          DEC     R3                      ;WE DONE ALL WORDS?
030302' 001367                          BNE     70$                     ;NOT YET
030304'                          ENDSUB ;#2
030304'                                                                 L10125:
```

```
030304' 104403                                                                     TRAP    C$ESUB
                                   ;
                                   ;CKECK TO SEE IF ALL PATTERNS HAVE BEEN RUN THROUGH
                                   ;
030306' 062704  000002                     ADD     #2,R4                  ;POINT TO NEXT DATA PATTERN
030312' 020427  000532'                    CMP     R4,#PAT6               ;WE DONE ALL DATA PATTERNS?
030316' 001402                             BEQ     90$                    ;YES END OF TEST
030320' 000137  027544'                    JMP     T11.1                  ;NO CONTINUE WITH NEW DATA PATTERN
030324'                            90$:
030324'                            ENDTST
030324'                                                                L10123:
030324' 104401                                                                     TRAP    C$ETST
```

```
                                        .SBTTL  TEST 12: LOAD AND START FUNCTION TEST

                                        ;*****************************************************************************
                                        ;
                                        ;THIS TEST WILL VERIFY THAT THE LOAD AND START MICROADDRESS PORT COMMAND
                                        ;IS OPERATIONAL.
                                        ;
                                        ;THE PROCESS IS TO LOAD WCS WITH MICROCODE THAT WHEN STARTED WILL WRITE
                                        ;A PATTERN OF DATA TO THE LITE-BYTE FIELD OF PCSR1 REGISTER WHICH CAN BE READ
                                        ;FROM THE UNIBUS AND BE VERIFIED.
                                        ;
                                        ;NOTE: THIS TEST USES MICROCODE MODULE 'A'
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-FORMAT UNIBUS DATA BLOCK WITH BYTE COUNT, WCS DESTINATION ADDRESS
                                        ;         AND UNIBUS SOURCE ADDRESS OF THE MICROCODE.
                                        ;       2-WRITE PORT CONTROL BLOCK WITH ADDESS OF UNIBUS DATA BLOCK AND
                                        ;         LOAD INTERNAL MEMORY FUNCTION CODE.
                                        ;       3-WRITE PCSR2 AND PCSR3 WITH ADDRESS OF THE PORT CONTROL BLOCK.
                                        ;       4-WRITE PCSR0 WITH <GET PCB> PORT COMMAND
                                        ;       5-READ PCSR0 AND VERIFY 'DNI'
                                        ;       6-WRITE PCSR0 'DNI' TO CLEAR
                                        ;       7-WRITE PCSR0 WITH <GET COMMAND> PORT COMMAND
                                        ;       8-READ PCSR0 AND VERIFY 'DNI'
                                        ;       9-WRITE PCSR0 'DNI' TO CLEAR
                                        ;       10-WRITE PORT CONTROL BLOCK WITH START ADDRESS OF WCS MICROCODE
                                        ;          AND WITH LOAD AND START FUNCTION CODE
                                        ;       11-WRITE PCSR0 WITH <GET COMMAND> PORT COMMAND
                                        ;       12-READ PCSR0 AND VERIFY 'DNI' SET
                                        ;       13-READ PCSR1 AND VERIFY BITS 13:8 ARE PATTERN WRITTEN BY MICROCODE
                                        ;       14-ISSUE RESET TO PCSR0 TO RESTART OPERATIONAL MICROCODE
                                        ;       15-READ PCSR0 AND VERIFY 'DNI' SET
                                        ;       16-WRITE PCSR0 WITH 'DNI' TO CLEAR
                                        ;
                                        ;*****************************************************************************

 030326'                                BGNTST
 030326'                                                                        T12::
                                        ;
                                        ;LOAD MICROCODE MODULE 'A' INTO THE TOP HALF OF WCS
                                        ;
 030326'                                 BGNSEG
 030326' 104404                                                                         TRAP    C$BSEG
                                        ;
                                        ;FORMAT THE UNIBUS DATA BLOCK AND THE PORT CONTROL BLOCK FOR THE LOAD
                                        ;INTERNAL MEMORY FUNCTION
                                        ;
                                                                        ;SETUP UDBB
 030330' 013737  000000G 000616'        MOV     MICASZ,UDBB             ;SIZE OF MICROCODE MODULE TO LOAD
 030336' 012737  000000G 000620'        MOV     #MICROA,UDBB+2          ;BASE ADDRESS OF MICROCODE MODULE
 030344' 005037  000622'                CLR     UDBB+4
 030350' 012737  010000  000624'        MOV     #WCSADR+<WCSSIZ/2>,UDBB+6 ;LOAD INTO TOP HALF OF WCS

                                                                        ;SETUP PCB
 030356' 012737  000021  000606'        MOV     #LIM,PCBB               ;'LOAD INTERNAL MEMORY' FUNCTION
```

```
030364' 012737  000616' 000610'        MOV     #UDBB,PCBB+2            ;SET ADDRESS OF UDBB
030372' 005037  000612'                CLR     PCBB+4
030376' 012777  000606' 147736         MOV     #PCBB,@PCSR2            ;TELL DEUNA WHERE PCBB IS
030404' 005077  147734                 CLR     @PCSR3

030410' 012777  000001  147720         MOV     #GETPCB,@PCSR0          ;ISSUE 'GET PCB' PORT COMMAND
030416' 012737  000176  000332'        MOV     #2*SECOND,METER         ;SETUP TIMER
030424' 004737  017316'                JSR     PC,CHKDNI               ;WAIT FOR 'DNI' TO SET
030430' 103022                         BCC     10$                     ;OK
                                                                       ;ERROR DNI NOT SET!
030432' 012737  001000' 000310'        MOV     #$DNI,BITNAM            ;SETUP ERROR MESSAGE
030440' 012737  001277' 000312'        MOV     #$NSET,BITSTA
030446' 012737  001342' 000314'        MOV     #$AFTER,PWHEN
030454' 012737  001350' 000316'        MOV     #$GTPCB,PCOMND
030462'                                ERRHRD  042,LASFT,MSG1          ;PRINT ERROR MESSAGE
030462' 104456                                                                            TRAP    C$ERHRD
030464' 000052                                                                            .WORD   42
030466' 002026'                                                                           .WORD   LASFT
030470' 012716'                                                                           .WORD   MSG1
030472'                                ESCAPE TST                      ;NO POINT IN CONTINUING TEST
030472' 104410                                                                            TRAP    C$ESCAPE
030474' 000310                                                                            .WORD   L10126-.
030476'                         10$:
030476'                           ENDSEG
030476'                                                                         10000$:
030476' 104405                                                                            TRAP    C$ESEG
030500'                           BGNSEG
030500' 104404                                                                            TRAP    C$BSEG
                                ;
                                ;WRITE ONE TO CLEAR 'DNI'
                                ;
030502' 004737  017362'                JSR     PC,CLRDNI               ;GO CLEAR 'DNI'
030506' 103006                         BCC     20$                     ;OK
                                                                       ;ERROR 'DNI' NOT CLEAR!
030510'                                ERRHRD  043,LASFT,RACMG7        ;PRINT ERROR MESSAGE
030510' 104456                                                                            TRAP    C$ERHRD
030512' 000053                                                                            .WORD   43
030514' 002026'                                                                           .WORD   LASFT
030516' 012670'                                                                           .WORD   RACMG7
030520'                                ESCAPE  TST                     ;DO NOT CONTINUE TEST
030520' 104410                                                                            TRAP    C$ESCAPE
030522' 000262                                                                            .WORD   L10126-.
030524'                         20$:
030524'                           ENDSEG
030524'                                                                         10001$:
030524' 104405                                                                            TRAP    C$ESEG
030526'                           BGNSEG
030526' 104404                                                                            TRAP    C$BSEG
                                ;
                                ;NOW THAT THE UNA KNOWS WHERE THE MICROCODE IS, ISSUE THE GET COMMAND PORT
                                ;COMMAND SO THE MICROCODE CAN GET LOADED INTO WCS
                                ;
030530' 012777  000002  147600         MOV     #GETCMD,@PCSR0          ;ISSUE <GET COMMAND> PORT COMMAND
030536' 012737  000176  000332'        MOV     #2*SECOND,METER         ;SETUP TIMER
030544' 004737  017316'                JSR     PC,CHKDNI               ;WAIT FOR 'DNI'
030550' 103022                         BCC     30$                     ;OK
```

```
                                                                        ;ERROR 'DNI' NOT SET!
030552' 012737  001000' 000310'         MOV     #SDNI,BITNAM            ;SETUP ERROR MESSAGE
030560' 012737  001277' 000312'         MOV     #SNSET,BITSTA
030566' 012737  001342' 000314'         MOV     #SAFTER,PWHEN
030574' 012737  001357' 000316'         MOV     #SGTCMD,PCOMND
030602'                                 ERRHRD  044,LASFT,MSG1          ;PRINT ERROR MESSAGE
030602' 104456                                                                  TRAP    C$ERHRD
030604' 000054                                                                  .WORD   44
030606' 002026'                                                                 .WORD   LASFT
030610' 012716'                                                                 .WORD   MSG1
030612'                                 ESCAPE TST                      ;LEAVE TEST
030612' 104410                                                                  TRAP    C$ESCAPE
030614' 000170                                                                  .WORD   L10126-.
030616'                         30$:
030616'                           ENDSEG
030616'                                                         10002$:
030616' 104405                                                                  TRAP    C$ESEG
030620'                           BGNSEG
030620' 104404                                                                  TRAP    C$BSEG

                                ;
                                ;WRITE ONE TO CLEAR 'DNI'
                                ;
030622' 004737  017362'                 JSR     PC,CLRDNI               ;GO CLEAR 'DNI'
030626' 103006                          BCC     40$                     ;OK
                                                                        ;ERROR 'DNI' NOT CLEAR
030630'                                 ERRHRD  045,LASFT,RACMG7        ;PRINT ERROR MESSAGE
030630' 104456                                                                  TRAP    C$ERHRD
030632' 000055                                                                  .WORD   45
030634' 002026'                                                                 .WORD   LASFT
030636' 012670'                                                                 .WORD   RACMG7
030640'                                 ESCAPE TST                      ;DO NOT CONTINUE TEST
030640' 104410                                                                  TRAP    C$ESCAPE
030642' 000142                                                                  .WORD   L10126-.
030644'                         40$:
030644'                           ENDSEG
030644'                                                         10003$:
030644' 104405                                                                  TRAP    C$ESEG
                                ;
                                ;OK, MICROCODE MODULE 'A' IS LOADED INTO WCS. NOW START IT AND CHECK PCSR1.
                                ;
030646'                           BGNSEG
030646' 104404                                                                  TRAP    C$BSEG
030650' 012737  000001  000606'         MOV     #LASM,PCBB              ;LOAD PCBB WITH 'LOAD AND START' FUNCTION CODE
030656' 012737  010000  000610'         MOV     #WCSADR+<WCSSIZ/2>,PCBB+2 ;STARTING MICROADDRESS
030664' 012777  000002  147444          MOV     #GETCMD,@PCSR0          ;ISSUE <GET COMMAND> PORT COMMAND
030672' 012737  000176  000332'         MOV     #2*SECOND,METER         ;SETUP TIMER
030700' 004737  017316'                 JSR     PC,CHKDNI               ;GO WAIT FOR 'DNI' TO SET
030704' 103020                          BCC     50$                     ;OK
                                                                        ;ERROR 'DNI' NOT SET!
030706' 012737  001000' 000310'         MOV     #SDNI,BITNAM            ;SETUP ERROR MESSAGE
030714' 012737  001277' 000312'         MOV     #SNSET,BITSTA
030722' 012737  001342' 000314'         MOV     #SAFTER,PWHEN
030730' 012737  001357' 000316'         MOV     #SGTCMD,PCOMND
030736'                                 ERRHRD  046,LASFT,MSG1          ;PRINT ERROR MESSAGE
030736' 104456                                                                  TRAP    C$ERHRD
030740' 000056                                                                  .WORD   46
```

```
030742' 002026'                                                                .WORD   LASFT
030744' 012716'                                                                .WORD   MSG1
030746' 017702  147366   50$:   MOV     @PCSR1,R2               ;GET PCSR1 CONTENTS
030752' 042702  140370          BIC     #^C<SFT!PSTATE>,R2      ;CLEAR ALL BUT SELF TEST AND STATE BITS
030756' 012701  012402          MOV     #SFTB0!SFTB2!SFTB4!INTST,R1 ;PATTERN THAT SHOULD BE IN PCSR1
030762' 020102                  CMP     R1,R2                   ;IS PCSR1 PATTERN CORRECT?
030764' 001404                  BEQ     60$                     ;YES
030766'                         ERRHRD  047,LASFT,MSG6          ;NO, PRINT ERROR MESSAGE
030766' 104456                                                                 TRAP    C$ERHRD
030770' 000057                                                                 .WORD   47
030772' 002026'                                                                .WORD   LASFT
030774' 013174'                                                                .WORD   MSG6
030776'                  60$:
030776'                    ENDSEG
030776'                                                            10004$:
030776' 104405                                                                 TRAP    C$ESEG
                         ;
                         ;EVERYTHING WORKED JUST FINE, NOW WE HAVE TO GET THE OPERATIONAL MICROCODE
                         ;GOING AGAIN BEFORE WE LEAVE OTHERWISE EVERYTHING WILL BE SCREWED UP.
                         ;
031000' 004737  020166'         JSR     PC,REUNA                ;RESET DEUNA TO RESTORE OPERATIONAL
                                                                ;MICROCODE
031004'                  ENDTST
031004'                                                            L10126:
031004' 104401                                                                 TRAP    C$ETST
```

```
                                        .SBTTL  TEST 13: COMPREHENSIVE WCS MEMORY TEST

                                        ;*******************************************************************8
                                        ;
                                        ;THIS TEST WILL EXHAUSTIVELY TEST THE WCS MEMORY.
                                        ;CUSTOM MICROCODE MODULE B, MICROTEST #1 IS USED TO DO THE ACTUAL TESTING.
                                        ;MICROTEST #1 RUNS A SERIES OF MICROSUBTESTS TESTS ON THE WCS MEMORY CHECKING
                                        ;FOR BOTH ADDRESS AND DATA ERRORS. IF AN ERROR DOES OCCUR THE PORT CONTROL
                                        ;BLOCK WILL CONTAIN THE INFORMATION ABOUT THE ERROR.
                                        ;
                                        ;PCBB+0: CONTAINS THE MICROSUBTEST THAT FAILED
                                        ;PCBB+1: 0 = DATA ERROR, 1 = ADDRESS ERROR
                                        ;PCBB+2: CONTAINS THE ADDRESS OF THE LOCATION
                                        ;PCBB+4: CONTAINS THE DATA THAT WAS WRITTEN
                                        ;PCBB+6: CONTAINS THE DATA THAT WAS READ
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROMODULE 'B' INTO THE TOP HALF OF WCS IF NOT ALREADY DONE SO
                                        ;       2-WAIT FOR THE MICROMONITOR TO BECOME ACTIVE
                                        ;       3-CLEAR PCBB LOCATIONS 0-7
                                        ;       4-TELL MICROMONITOR TO EXECUTE MICROTEST #1
                                        ;       5-WAIT FOR 'DNI'
                                        ;       6-IF ERROR PRINT PCBB CONTENTS
                                        ;       7-WRITE ONE TO CLEAR 'DNI'
                                        ;       8-RESTORE OPERATIONAL MICROCODE
                                        ;
                                        ;*******************************************************************
        031006'                         BGNTST
        031006'                                                                 T13::
                                        ;
                                        ;CHECK TO SEE IF MICROCODE MODULE 'B' HAS BEEN LOADED. IF NOT, LOAD IT INTO
                                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
        031006'                          BGNSEG
        031006' 104404                                                                  TRAP    C$BSEG
        031010' 022737  000102  000326'         CMP     #'B,MICRO               ;HAS MICROCODE MODULE 'B' BEEN LOADED?
        031016' 001004                          BNE     5$                      ;NO
        031020' 122777  000001  147312          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
        031026' 001435                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
        031030' 012737  000102  000326' 5$:     MOV     #'B,MICRO               ;GO LOAD MICROCODE MODULE B
        031036' 004737  020340'                 JSR     PC,LODMIC
        031042' 103564                          BCS     55$                     ;ERROR
        031044' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
        031052' 004737  017316'                 JSR     PC,CHKDNI
        031056' 103021                          BCC     20$                     ;OK
        031060' 012737  001000' 000310'         MOV     #$DNI,BITNAM
        031066' 012737  001277' 000312'         MOV     #$NSET,BITSTA
        031074' 012737  001342' 000314'         MOV     #$AFTER,PWHEN
        031102' 012737  001357' 000316'         MOV     #$GTCMD,PCOMND
        031110'                                 ERRHRD  048,WCSMEM,MSG1         ;PRINT ERROR MESSAGE
        031110' 104456                                                                  TRAP    C$ERHRD
        031112' 000060                                                                  .WORD   48
        031114' 002072'                                                                 .WORD   WCSMEM
        031116' 012716'                                                                 .WORD   MSG1
```

```
031120' 000535                          BR      55$
031122' 004737  017362'         20$:    JSR     PC,CLRDNI                       ;CLEAR DNI
031126' 103005                          BCC     25$
031130'                                 ERRHRD  049,WCSMEM,RACMG7               ;DNI DID NOT CLEAR!
031130' 104456                                                                          TRAP    C$ERHRD
031132' 000061                                                                          .WORD   49
031134' 002072'                                                                         .WORD   WCSMEM
031136' 012670'                                                                         .WORD   RACMG7
031140' 000525                          BR      55$
031142'                         25$:
031142'                          ENDSEG
031142'                                                                         10000$:
031142' 104405                                                                          TRAP    C$ESEG

                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE THEN START MICROTEST #1.
                                ;WAIT FOR 'DNI'. CHECK FOR ILLEGAL INTERRUPTS CHECK THE STATE BITS FOR AN
                                ;ERROR CONDITION. IF ERROR REPORT IT. WRITE '1' TO CLEAR 'DNI' AND RESTORE
                                ;OPERATIONAL MICROCODE.
                                ;
031144'                          BGNSEG
031144' 104404                                                                          TRAP    C$BSEG
031146' 004737  020060'                 JSR     PC,CHKMON                       ;WAIT FOR MICROMONITOR
031152' 103006                          BCC     30$                             ;OK
031154'                                 ERRHRD  050,WCSMEM,MSG46                ;PRINT ERROR
031154' 104456                                                                          TRAP    C$ERHRD
031156' 000062                                                                          .WORD   50
031160' 002072'                                                                         .WORD   WCSMEM
031162' 016666'                                                                         .WORD   MSG46
031164'                                 ESCAPE  TST                             ;LEAVE TEST
031164' 104410                                                                          TRAP    C$ESCAPE
031166' 000234                                                                          .WORD   L10127-.
031170' 012777  000606' 147144  30$:    MOV     #PCBB,@PCSR2                    ;TELL MICROCODE TEST WHERE PCBB IS
031176' 005077  147142                  CLR     @PCSR3
031202' 005037  000606'                 CLR     PCBB+0                          ;CLEAR OUT THE PCBB
031206' 005037  000610'                 CLR     PCBB+2
031212' 005037  000612'                 CLR     PCBB+4
031216' 005037  000614'                 CLR     PCBB+6
031222' 012777  000001  147106          MOV     #1,@PCSR0                       ;TELL MICROMONITOR TO EXECUTE...
                                                                                ;MICROTEST #1
031230' 012737  000770  000332'         MOV     #10*SECOND,METER                ;PUT SOME TIME ON THE METER
031236' 004737  017316'                 JSR     PC,CHKDNI                       ;WAIT FOR MICROTEST TO FINISH
031242' 103017                          BCC     40$                             ;OK, IT FINISHED
031244' 004737  020132'                 JSR     PC,CHKINT                       ;SEE IF ANY ERROR INTERRUPTS OCCURRED
031250' 103005                          BCC     35$                             ;NO, OK
031252'                                 ERRHRD  051,WCSMEM,MSG44                ;PRINT ERROR MESSAGE
031252' 104456                                                                          TRAP    C$ERHRD
031254' 000063                                                                          .WORD   51
031256' 002072'                                                                         .WORD   WCSMEM
031260' 016442'                                                                         .WORD   MSG44
031262' 000454                          BR      55$                             ;LEAVE
031264' 012702  000003          35$:    MOV     #3,R2                           ;MICROTEST # THAT IS HUNG
031270'                                 ERRHRD  052,WCSMEM,MSG12                ;TELL MICROTEST HUNG
031270' 104456                                                                          TRAP    C$ERHRD
031272' 000064                                                                          .WORD   52
031274' 002072'                                                                         .WORD   WCSMEM
031276' 013466'                                                                         .WORD   MSG12
```

```
031300' 000445                            BR      55$
031302' 122777  000003  147030  40$:      CMPB    #INERR,@PCSR1            ;DID AN ERROR OCCUR?
031310' 001032                            BNE     50$                      ;NO
031312' 122737  000000  000607'           CMPB    #DATERR,PCBB+1           ;YES, WAS IT A DATA ERROR?
031320' 001003                            BNE     45$                      ;NO
031322' 012702  001446'                   MOV     #$DATER,R2               ;YES, POINT TO DATA ERROR STRING
031326' 000411                            BR      46$
031330' 122737  000001  000607' 45$:      CMPB    #ADRERR,PCBB+1           ;WAS IT AN ADDRESS ERROR
031336' 001003                            BNE     47$                      ;NO
031340' 012702  001461'                   MOV     #$ADRER,R2               ;POINT TO ADDRESS ERROR STRING
031344' 000402                            BR      46$
031346' 012702  001502'         47$:      MOV     #$PARER,R2               ;POINT TO PARITY ERROR STRING
031352' 013701  000610'         46$:      MOV     PCBB+2,R1                ;GET FAILING ADDRESS
031356' 013703  000612'                   MOV     PCBB+4,R3                ;GET GOOD DATA
031362' 013704  000614'                   MOV     PCBB+6,R4                ;GET BAD DATA
031366'                                   ERRHRD  053,WCSMEM,MSG16         ;PRINT ERROR MESSAGE
031366' 104456                                                                      TRAP    C$ERHRD
031370' 000065                                                                      .WORD   53
031372' 002072'                                                                     .WORD   WCSMEM
031374' 013636'                                                                     .WORD   MSG16
031376' 004737  017362'         50$:      JSR     PC,CLRDNI                ;CLEAR DNI
031402' 103004                            BCC     55$
031404'                                   ERRHRD  054,WCSMEM,RACMG7        ;DNI DID NOT CLEAR!
031404' 104456                                                                      TRAP    C$ERHRD
031406' 000066                                                                      .WORD   54
031410' 002072'                                                                     .WORD   WCSMEM
031412' 012670'                                                                     .WORD   RACMG7
031414' 004737  020166'         55$:      JSR     PC,REUNA                 ;RESTORE OPERATIONAL MICROCODE
031420'                                 ENDSEG
031420'                                                                     10001$:
031420' 104405                                                                      TRAP    C$ESEG
031422'                                 ENDTST
031422'                                                                     L10127:
031422' 104401                                                                      TRAP    C$ETST
```

```
                                        .SBTTL  TEST 14: INTERRUPT VECTOR TEST

                                        ;*********************************************************************
                                        ;
                                        ;THIS TEST WILL VERIFY THAT THE INTERRUPT INTERFACE LOGIC OF THE DEUNA
                                        ;IS CAPABLE OF GENERATING AN INTERUPT VECTOR AND ARBITRATING FOR CONTROL
                                        ;OF THE UNIBUS.
                                        ;THE DEUNA INTERRUPT ENABLE BIT WILL BE SET AND AN INTERRUPT WILL BE
                                        ;CAUSED BY ISSUING A NOP PORT COMMAND. AN INTERRUPT IS EXPECTED AT THE
                                        ;CORRECT VECTOR AND AT THE CORRECT PRIORITY.
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-SETUP INTERRUPT VECTOR
                                        ;       2-CLEAR INTERRUPT FLAG
                                        ;       3-SET INTERRUPT ENABLE IN PCSRO
                                        ;       4-ISSUE NOP PORT COMMAND
                                        ;       5-WAIT FOR DNI SET IN PCSRO
                                        ;       6-VERIFY INTERRUPT FLAG SET
                                        ;       7-VERIFY INTERRUPT AT CORRECT PRIORITY
                                        ;       8-RELEASE INTERRUPT VECTOR
                                        ;       9-CLEAR INTERRUPT ENABLE IN PCSRO
                                        ;       10-WRITE DNI TO CLEAR
                                        ;
                                        ;*********************************************************************

 031424'                                BGNTST
 031424'                                                                        T14::
                                        ;
                                        ;SETUP INTERRUPT VECTOR, CLEAR INTERRUPT FLAG, SET INTERRUPT ENABLE,
                                        ;ISSUE NOP PORT COMMAND AND WAIT FOR 'DNI'
                                        ;
 031424'                                  BGNSEG
 031424' 104404                                                                         TRAP    C$BSEG
 031426'                                        SETVEC  UNAVEC,#UNASRV,UNAPRI   ;SETUP DEUNA INTERRUPT VECTOR
 031426' 013746  000272'                                                                MOV     UNAPRI,-(SP)
 031432' 012746  022044'                                                                MOV     #UNASRV,-(SP)
 031436' 013746  000270'                                                                MOV     UNAVEC,-(SP)
 031442' 012746  000003                                                                 MOV     #3,-(SP)
 031446' 104437                                                                         TRAP    C$SVEC
 031450' 062706  000010                                                                 ADD     #10,SP
 031454' 005037  000672'                        CLR     UNAINT                  ;CLEAR UNA INTERRUPTED FLAG
 031460' 012777  000100  146650                 MOV     #IE,@PCSRO              ;SET INTERRUPT ENABLE
 031466' 012777  000106  146642                 MOV     #IE!PNOP,@PCSRO         ;ISSUE NOP PORT COMMAND
 031474' 012737  000176  000332'                MOV     #2*SECOND,METER         ;SETUP TIMER
 031502' 004737  017316'                        JSR     PC,CHKDNI               ;GO WAIT FOR DNI
 031506' 103020                                 BCC     10$                     ;OK
                                                                                ;ERROR DNI NOT SET!
 031510' 012737  001000' 000310'                MOV     #$DNI,BITNAM            ;SETUP ERROR MESSAGE
 031516' 012737  001277' 000312'                MOV     #$NSET,BITSTA
 031524' 012737  001342' 000314'                MOV     #$AFTER,PWHEN
 031532' 012737  001405' 000316'                MOV     #$NOP,PCOMND
 031540'                                        ERRHRD  055,INTVEC,MSG1         ;PRINT ERROR MESSAGE
 031540' 104456                                                                         TRAP    C$ERHRD
 031542' 000067                                                                         .WORD   55
 031544' 002121'                                                                        .WORD   INTVEC
 031546' 012716'                                                                        .WORD   MSG1
```

```
031550'                         10$:
031550'                           ENDSEG
031550'                                                                         10000$:
031550' 104405                                                                          TRAP    C$ESEG
                                ;
                                ;VERIFY THAT INTERRUPT OCCURRED AT CORRECT PRIORITY
                                ;
031552'                           BGNSEG
031552' 104404                                                                          TRAP    C$BSEG
031554' 012701  000006                  MOV     #6,R1                   ;START CPU PRIORITY LOWERING
                                                                        ;FROM PRIORITY 7
031560' 010102                  20$:    MOV     R1,R2                   ;GET INTEGER PRIORITY
031562' 072227  000005                  ASH     #5,R2                   ;PUT PRIORITY IN CORRECT POSITION
031566'                                 SETPRI  R2                      ;SET NEW PRIORITY
031566' 010200                                                                          MOV     R2,R0
031570' 104441                                                                          TRAP    C$SPRI
031572' 000240                          NOP                             ;LET INTERRUPT OCCUR HERE IF PENDING
031574' 005301                          DEC     R1                      ;LOWER PRIORITY
031576' 100370                          BPL     20$                     ;IF DONE FROM 6-->0 THEN DONE

031600' 005737  000672'                 TST     UNAINT                  ;UNA SHOULD HAVE INTERRUPTED BY NOW
031604' 001006                          BNE     30$                     ;OK, GO CHECK THE PRIORITY OF THE INTERRUPT
                                                                        ;ERROR! UNA DID NOT INTERRUPT
031606'                                 ERRHRD  056,INTVEC,MSG7         ;PRINT ERROR MESSAGE
031606' 104456                                                                          TRAP    C$ERHRD
031610' 000070                                                                          .WORD   56
031612' 002121'                                                                         .WORD   INTVEC
031614' 013262'                                                                         .WORD   MSG7
031616'                                 ESCAPE TST                      ;LEAVE TEST
031616' 104410                                                                          TRAP    C$ESCAPE
031620' 000072                                                                          .WORD   L10130-.
031622'                         30$:
031622' 013701  000272'                 MOV     UNAPRI,R1               ;GET UNA PRIORITY
031626' 072127  177773                  ASH     #-5,R1                  ;MAKE IT AN INTEGER
031632' 005301                          DEC     R1                      ;THIS IS THE CPU PRIORITY WHEN THE...
031634' 072127  000005                  ASH     #5,R1                   ;UNA SHOULD HAVE INTERRUPTED
031640' 020137  000676'                 CMP     R1,CPUPRI               ;DID UNA INTERRUPT AT CORRECT PRIORITY?
031644' 001404                          BEQ     40$                     ;YES,
                                                                        ;ERROR! UNA INTERRUPT PRIORITY INCORRECT
031646'                                 ERRHRD  057,INTVEC,MSG8 ;PRINT ERROR MESSAGE
031646' 104456                                                                          TRAP    C$ERHRD
031650' 000071                                                                          .WORD   57
031652' 002121'                                                                         .WORD   INTVEC
031654' 013310'                                                                         .WORD   MSG8
031656'                         40$:
031656'                           ENDSEG
031656'                                                                         10001$:
031656' 104405                                                                          TRAP    C$ESEG
031660'                                 CLRVEC  UNAVEC                  ;RELEASE INTERRUPT VECTOR
031660' 013700  000270'                                                                 MOV     UNAVEC,R0
031664' 104436                                                                          TRAP    C$CVEC
031666' 012777  000000  146442          MOV     #0,@PCSR0               ;CLEAR INTERRUPT ENABLE
031674' 004737  017362'                 JSR     PC,CLRDNI               ;CLEAR DNI
031700' 103004                          BCC     50$
031702'                                 ERRHRD  058,INTVEC,RACMG7       ;ERROR! DNI DID NOT CLEAR
031702' 104456                                                                          TRAP    C$ERHRD
```

```
031704' 000072                                          .WORD   58
031706' 002121'                                         .WORD   INTVEC
031710' 012670'                                         .WORD   RACMG7
031712'                 50$:
031712'                 ENDTST
031712'                                       L10130:
031712' 104401                                          TRAP    C$ETST
```

```
                                        .SBTTL  TEST 15: PCSRO INTERRUPT BIT TEST

                                        ;*************************************************************************
                                        ;
                                        ;THIS TEST WILL VERIFY THAT EACH OF THE INTERRUPT BITS IN REGISTER PCSRO
                                        ;CAN CAUSE AN INTERRUPT.
                                        ;
                                        ;EACH OF THE INTERRUPTS OF REGISTER PCSRO IS SET UNDER THE CONTROL OF THE
                                        ;T11 AND NOT DIRECTLY BY HARDWARE. THE T11 THEREFORE CAN INITIATE UNIBUS
                                        ;INTERRUPTS BY SETTING BITS IN REGISTER PCSRO.
                                        ;
                                        ;THIS TEST USES MICROMODULE C, MICROTEST #1.
                                        ;MICROCODE MODULE C IS LOADED IF NOT ALREADY DONE SO BY A PREVIOUS TEST.
                                        ;
                                        ;THE DEUNA INTERRUPT VECTOR IS SETUP TO STORE THE CONTENTS OF PCSRO WHEN THE
                                        ;INTERRUPT OCCURS. PCBB+0 IS LOADED WITH THE INTERRUPT BIT THAT IS TO BE TESTED
                                        ;THEN PCSRO COMMAND BITS ARE LOADED WITH A 1 TO TELL THE T11 TO EXECUTE
                                        ;MICROTEST #1. WE WAIT FOR THE INTERRUPT TO OCCUR THEN SEE IF THE CONTENTS
                                        ;OF PCSRO AT THE TIME OF THE INTERRUPT CONTAINED THE CORRECT INTERRUPT BIT.
                                        ;THE TEST IS REPEATED FOR ALL THE INTERRUPT BITS.
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROMODULE C INTO THE TOP HALF OF WCS IF NOT ALREADY DONE SO
                                        ;       2-SETUP DEUNA INTERRUPT VECTOR
                                        ;       3-WAIT FOR THE MICROMONITOR TO BECOME ACTIVE
                                        ;       4-SET A BIT IN PCBB+0 THAT CORRESPONDS TO THE INTERRUPT BIT TO TEST
                                        ;       5-SET INTERRUPT ENABLE
                                        ;       6-TELL MICROMONITOR TO EXECUTE MICROTEST #1
                                        ;       7-VERIFY INTERRUPT OCCURRED
                                        ;       8-VERIFY CORRECT BIT CAUSED INTERRUPT
                                        ;       9-WRITE ONE TO CLEAR INTERRUPT BIT
                                        ;       10-REPEAT STEPS 3-9 FOR ALL THE INTERRUPT BITS
                                        ;
                                        ;*************************************************************************

 031714'                                BGNTST
 031714'                                                                        T15::
                                        ;
                                        ;CHECK TO SEE IF MODULE 'C' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
 031714'                                  BGNSEG
                                                                                        TRAP    C$BSEG
 031714' 104404
 031716' 022737  000103  000326'        CMP     #'C,MICRO               ;HAS MICROCODE MODULE C BEEN LOADED?
 031724' 001004                         BNE     5$                      ;NO
 031726' 122777  000001  146404         CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
 031734' 001440                         BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
 031736' 012737  000103  000326' 5$:    MOV     #'C,MICRO               ;SETUP TO LOAD MODULE C
 031744' 004737  020340'                JSR     PC,LODMIC               ;GO LOAD MICROMODULE C
 031750' 103002                         BCC     10$                     ;SUCCESS
 031752'                                ESCAPE  TST                     ;ERROR OCCURRED LEAVE
                                                                                        TRAP    C$ESCAPE
 031752' 104410
                                                                                        .WORD   L10131-.
 031754' 000444
 031756' 012737  000176  000332' 10$:   MOV     #2*SECOND,METER         ;PUT SOME TIME ON THE METER
 031764' 004737  017316'                JSR     PC,CHKDNI               ;GO WAIT FOR DNI TO SET AFTER THE
```

```
                                                                  ;LOAD AND START FUNCTION
031770' 103022                              BCC     20$                  ;OK
031772' 012737  001000' 000310'             MOV     #$DNI,BITNAM         ;SETUP ERROR MESSAGE
032000' 012737  001277' 000312'             MOV     #$NSET,BITSTA
032006' 012737  001342' 000314'             MOV     #$AFTER,PWHEN
032014' 012737  001357' 000316'             MOV     #$GTCMD,PCOMND
032022'                                     ERRHRD  059,INTBIT,MSG1      ;PRINT ERROR MESSAGE
032022' 104456                                                                                TRAP    C$ERHRD
032024' 000073                                                                                .WORD   59
032026' 002156'                                                                               .WORD   INTBIT
032030' 012716'                                                                               .WORD   MSG1
032032'                                     ESCAPE TST
032032' 104410                                                                                TRAP    C$ESCAPE
032034' 000364                                                                                .WORD   L10131-.
032036' 004737  017362'            20$:     JSR     PC,CLRDNI            ;CLEAR DNI
032042' 103006                              BCC     25$
032044'                                     ERRHRD  060,INTBIT,RACMG7    ;ERROR! DNI DID NOT CLEAR
032044' 104456                                                                                TRAP    C$ERHRD
032046' 000074                                                                                .WORD   60
032050' 002156'                                                                               .WORD   INTBIT
032052' 012670'                                                                               .WORD   RACMG7
032054'                                     ESCAPE TST
032054' 104410                                                                                TRAP    C$ESCAPE
032056' 000342                                                                                .WORD   L10131-.
032060'                            25$:     SETVEC  UNAVEC,#UNASRV,UNAPRI   ;SETUP DEUNA INTERRUPT VECTOR
032060' 013746  000272'                                                                       MOV     UNAPRI,-(SP)
032064' 012746  022044'                                                                       MOV     #UNASRV,-(SP)
032070' 013746  000270'                                                                       MOV     UNAVEC,-(SP)
032074' 012746  000003                                                                        MOV     #3,-(SP)
032100' 104437                                                                                TRAP    C$SVEC
032102' 062706  000010                                                                        ADD     #10,SP
032106'                                  ENDSEG
032106'                                                                              10000$:
032106' 104405                                                                                TRAP    C$ESEG
                                   ;
                                   ;THE FOLLOWING LOOP WILL BE EXECUTED 6 TIMES- ONCE FOR EACH BIT 10 THRU 15
                                   ;OF PCSR0. IT WAITS FOR THE MICROMONITOR TO BECOME ACTIVE THEN CALLS
                                   ;MICROTEST #1 TO SET A BIT IN PCSR0 AS DEFINED BY THE PARAMETER IN PCBB+0.
                                   ;THIS OPERATION SHOULD CAUSE AN INTERRUPT WHICH WILL BE REFLECTED BY THE
                                   ;VARIABLE 'UNAINT' WHICH IS SET BY THE UNA INTERRUPT SERVICE ROUTINE. UNAINT
                                   ;IS LOADED WITH THE VALUE OF PCSR0 AT THE TIME OF THE INTERRUPT. THIS WAY
                                   ;THE BIT THAT CAUSED THE INTERRUPT CAN BE CHECKED.
                                   ;
032110' 005037  000304'                     CLR     CSRNUM               ;CHECKING PCSR0
032114' 012737  001277' 000312'             MOV     #$NSET,BITSTA        ;CHECKING FOR SET BITS
032122' 012701  002000                      MOV     #BIT10,R1            ;START WITH BIT 10
032126' 012737  000012  000306'             MOV     #10.,BITNUM
032134'                            26$:
032134'                             BGNSEG
032134' 104404                                                                                TRAP    C$BSEG
032136' 004737  020060'                     JSR     PC,CHKMON            ;WAIT FOR MICROMONITOR
032142' 103006                              BCC     30$                  ;OK
032144'                                     ERRHRD  061,INTBIT,MSG46     ;PRINT ERROR
032144' 104456                                                                                TRAP    C$ERHRD
032146' 000075                                                                                .WORD   61
032150' 002156'                                                                               .WORD   INTBIT
```

```
032152' 016666'                                                                                .WORD   MSG46
032154'                                   ESCAPE  TST                    ;LEAVE TEST
032154' 104410                                                                                 TRAP    C$ESCAPE
032156' 000242                                                                                 .WORD   L10131-.
032160' 005037  000672'           30$:    CLR     UNAINT                 ;CLEAR UNA INTERRUPTED FLAG
032164' 010137  000606'                   MOV     R1,PCBB+0              ;TELL MICROMONITOR WHICH BIT TO SET
032170' 012737  000077  000332'           MOV     #1*SECOND,METER        ;PUT SOME TIME ON THE METER
032176' 012777  000100  146132            MOV     #IE,@PCSRO             ;SET INTERRUPT ENABLE AND...
                                                                         ;CLEAR SELF TEST BITS IN PCSR1
032204' 012777  000101  146124            MOV     #IE!1,@PCSRO           ;INVOKE INTERRUPT BIT MICROTEST
032212' 004737  017264'                   JSR     PC,TIMON               ;TURN ON THE TIMER
032216'                                   SETPRI  #PRI04                 ;LOWER PRIORITY TO ALLOW UNA INTERRUPT
032216' 012700  000200                                                                         MOV     #PRI04,R0
032222' 104441                                                                                 TRAP    C$SPRI
032224' 005737  000672'           35$:    TST     UNAINT                 ;HAS INTERRUPT OCCURRED YET?
032230' 001022                            BNE     40$                    ;YES
032232' 005737  000332'                   TST     METER                  ;NO, HAS METER EXPIRED?
032236' 001372                            BNE     35$                    ;NOT YET
032240' 004737  017302'                   JSR     PC,TIMOFF              ;TIMER HAS EXPIRED, SHUT IT OFF
                                                                         ;ERROR, NO UNA INTERRUPT!
032244' 013703  000306'                   MOV     BITNUM,R3              ;GET WHICH BIT
032250' 006303                            ASL     R3                     ;MAKE IT A BYTE OFFSET
032252' 012737  000360' 000310'           MOV     #BNAMTO,BITNAM         ;POINT TO TABLE OF BIT MNEMONICS
032260' 060337  000310'                   ADD     R3,BITNAM              ;INDEX INTO TABLE OF BIT MNEMONICS
032264'                                   ERRHRD  062,INTBIT,MSG9        ;PRINT ERROR MESSAGE
032264' 104456                                                                                 TRAP    C$ERHRD
032266' 000076                                                                                 .WORD   62
032270' 002156'                                                                                .WORD   INTBIT
032272' 013332'                                                                                .WORD   MSG9
032274' 000402                            BR      45$
032276' 004737  017302'           40$:    JSR     PC,TIMOFF              ;INTERRUPT OCCURRED, SHUT OFF THE TIMER
032302' 013703  000672'           45$:    MOV     UNAINT,R3              ;GET SAVED PCSRO CONTENTS
032306' 042703  000377                    BIC     #377,R3                ;CLEAR UNWANTED BITS
032312' 020103                            CMP     R1,R3                  ;DID CORRECT BIT CAUSE INTERRUPT?
032314' 001404                            BEQ     50$                    ;YES, OK
                                                                         ;ERROR, INCORRECT BIT CAUSED INTERRUPT!
032316'                                   ERRHRD  063,INTBIT,RACMG2      ;PRINT ERROR MESSAGE
032316' 104456                                                                                 TRAP    C$ERHRD
032320' 000077                                                                                 .WORD   63
032322' 002156'                                                                                .WORD   INTBIT
032324' 012526'                                                                                .WORD   RACMG2
032326'                           50$:
032326'                             ENDSEG
032326'                                                                             10001$:
032326' 104405                                                                                 TRAP    C$ESEG

                                  ;
                                  ;NOW WRITE '1' TO CLEAR THE BIT THAT CAUSED THE INTERRUPT
                                  ;
032330'                             BGNSEG
032330' 104404                                                                                 TRAP    C$BSEG
032332' 010377  146000                    MOV     R3,@PCSRO              ;WRITE '1' TO CLEAR INTERRUPT BIT
032336' 017704  145774                    MOV     @PCSRO,R4              ;READ IT BACK
032342' 030403                            BIT     R4,R3                  ;IS BIT CLEARED?
032344' 001420                            BEQ     55$                    ;YES
                                                                         ;ERROR, BIT DID NOT CLEAR!
032346' 012705  000017                    MOV     #15.,R5                ;
```

```
032352' 006303                  52$:    ASL     R3                      ;IS THIS THE BIT WE SET?
032354' 103402                          BCS     53$                     ;YES
032356' 005305                          DEC     R5                      ;NO
032360' 000774                          BR      52$
032362' 006305                  53$:    ASL     R5                      ;MAKE IT A BYTE OFFSET
032364' 012737  000360' 000310'         MOV     #BNAMTO,BITNAM          ;GET POINTER TO BIT NAME MNEMONICS
032372' 060537  000310'                 ADD     R5,BITNAM               ;INDEX INTO BIT NAME TABLE
032376'                                 ERRHRD  064,INTBIT,MSG33        ;PRINT ERROR MESSAGE
032376' 104456                                                                  TRAP    C$ERHRD
032400' 000100                                                                  .WORD   64
032402' 002156'                                                                 .WORD   INTBIT
032404' 015574'                                                                 .WORD   MSG33
032406'                         55$:
032406'                           ENDSEG
032406'                                                                 10002$:
032406' 104405                                                                  TRAP    C$ESEG
032410' 005237  000306'                 INC     BITNUM                  ;SETUP FOR NEXT BIT
032414' 006301                          ASL     R1                      ;POINT TO NEXT HIGHER BIT
032416' 103246                          BCC     26$                     ;IF BIT 15 NOT DONE YET GO ON
                                                                        ;ELSE ALL DONE
032420'                         ENDTST
032420'                                                                 L10131:
032420' 104401                                                                  TRAP    C$ETST
```

```
                                          .SBTTL  TEST 16: TIMER TEST

                                          ;**********************************************************************
                                          ;
                                          ;THIS TEST WILL USE THE CUSTOM MICROCODE MODULE 'C' TO CHECK THE OPERATION
                                          ;OF THE TIMER.
                                          ;THE TIMER IS ACCESSIBLE ONLY TO THE T11 PROCESSOR. THE HOST PROCESSOR
                                          ;CAN START THE TIMER ONLY WITH THE ASSISTANCE OF THE T11 PROCESSOR.
                                          ;
                                          ;FOR THIS TEST THE MICROCODE WILL BE LOADED ONLY IF IT HAS NOT ALREADY
                                          ;BEEN DONE BY A PREVIOUS TEST.
                                          ;
                                          ;WHEN THE MICROCODE IS STARTED THE T11 WILL START THE TIMER AND WILL
                                          ;SET 'DNI' WHEN THE TIMING INTERVAL HAS EXPIRED. THE INTERVAL IS 10 SECONDS.
                                          ;
                                          ;ANY TIME FROM 8 TO 12 SECONDS IS AN ACCEPTABLE RANGE.
                                          ;
                                          ;TEST SEQUENCE:
                                          ;       1-LOAD MICROCODE MODULE 'C' IF NOT ALREADY DONE SO
                                          ;       2-START MICROCODE
                                          ;       3-START TIMER
                                          ;       4-WAIT FOR 'DNI'
                                          ;       5-CHECK TIME INTERVAL
                                          ;
                                          ;**********************************************************************

 032422'                                  BGNTST
 032422'                                                                           T16::
                                          ;
                                          ;CHECK TO SEE IF MODULE 'C' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                          ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                          ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                          ;
 032422'                                   BGNSEG
 032422' 104404                                                                                  TRAP    C$BSEG
 032424' 022737  000103  000326'                  CMP     #'C,MICRO              ;HAS MICROCODE MODULE C BEEN LOADED?
 032432' 001004                                   BNE     5$                     ;NO
 032434' 122777  000001  145676                   CMPB    #INMON,@PCSR1          ;YES, IS THE MICROMONITOR ACTIVE?
 032442' 001440                                   BEQ     20$                    ;YES SKIP LOADING THE MICROMODULE
 032444' 012737  000103  000326'  5$:             MOV     #'C,MICRO              ;NO
 032452' 004737  020340'                          JSR     PC,LODMIC              ;GO LOAD MICROCODE MODULE C
 032456' 103002                                   BCC     10$                    ;OK
 032460'                                          ESCAPE TST                     ;ERROR OCCURRED LOADING MICROCODE
 032460' 104410                                                                                  TRAP    C$ESCAPE
 032462' 000254                                                                                  .WORD   L10132-.
 032464' 012737  000176  000332'  10$:            MOV     #2*SECOND,METER        ;WAIT FOR MICROMONITOR TO TAKE OVER
 032472' 004737  017316'                          JSR     PC,CHKDNI
 032476' 103022                                   BCC     20$
 032500' 012737  001000' 000310'                  MOV     #$DNI,BITNAM
 032506' 012737  001277' 000312'                  MOV     #$NSET,BITSTA
 032514' 012737  001342' 000314'                  MOV     #$AFTER,PWHEN
 032522' 012737  001357' 000316'                  MOV     #$GTCMD,PCOMND
 032530'                                          ERRHRD  065,TIMTST,MSG1
 032530' 104456                                                                                  TRAP    C$ERHRD
 032532' 000101                                                                                  .WORD   65
 032534' 002216'                                                                                 .WORD   TIMTST
```

```
032536' 012716'                                                            .WORD   MSG1
032540'                                 ESCAPE TST
032540' 104410                                                             TRAP    C$ESCAPE
032542' 000174                                                             .WORD   L10132-.
032544' 004737  017362'         20$:    JSR     PC,LLRDNI           ;CLEAR DNI
032550' 103006                          BCC     25$
032552'                                 ERRHRD  066,TIMTST,RACMG7   ;ERROR DNI DID NOT CLEAR!
032552' 104456                                                             TRAP    C$ERHRD
032554' 000102                                                             .WORD   66
032556' 002216'                                                            .WORD   TIMTST
032560' 012670'                                                            .WORD   RACMG7
032562'                                 ESCAPE TST
032562' 104410                                                             TRAP    C$ESCAPE
032564' 000152                                                             .WORD   L10132-.
032566'                         25$:
032566'                           ENDSEG
032566'                                                            10000$:
032566' 104405                                                             TRAP    C$ESEG

                                ;
                                ;WAIT FOR THE MICROCODE TO ENTER THE MICROMONITOR, SETUP OUR TIMEOUT TO BE
                                ;12 SECONDS (THIS GIVES A BETTER RESOLUTION THAN 1 SECOND), START THE MICROTEST
                                ;BY LOADING THE COMMAND FIELD OF PCSRO WITH THE MICROTEST # TO EXECUTE.
                                ;CHECK FOR 'DNI' TO BE SET IN LESS THAN 12 SECONDS.
                                ;
032570'                           BGNSEG
032570' 104404                                                             TRAP    C$BSEG
032572' 004737  020060'                 JSR     PC,CHKMON           ;WAIT FOR MICROMONITOR
032576' 103006                          BCC     30$                 ;OK
032600'                                 ERRHRD  068,TIMTST,MSG46    ;PRINT ERROR
032600' 104456                                                             TRAP    C$ERHRD
032602' 000104                                                             .WORD   68
032604' 002216'                                                            .WORD   TIMTST
032606' 016666'                                                            .WORD   MSG46
032610'                                 ESCAPE  TST                 ;LEAVE TEST
032610' 104410                                                             TRAP    C$ESCAPE
032612' 000124                                                             .WORD   L10132-.
032614' 012737  001356  000332' 30$:    MOV     #750.,METER         ;TIMEOUT = 12 SECONDS
032622' 012777  000002  145506          MOV     #2,@PCSRO           ;START MICROTEST #2
032630' 004737  017316'                 JSR     PC,CHKDNI           ;WAIT FOR DNI
032634' 103006                          BCC     40$                 ;OK IT FINISHED IN TIME
032636'                                 ERRHRD  069,TIMTST,MSG10    ;NO TIMER INTERRUPT
032636' 104456                                                             TRAP    C$ERHRD
032640' 000105                                                             .WORD   69
032642' 002216'                                                            .WORD   TIMTST
032644' 013360'                                                            .WORD   MSG10
032646'                                 ESCAPE TST
032646' 104410                                                             TRAP    C$ESCAPE
032650' 000066                                                             .WORD   L10132-.
                                ;
                                ;OK THE TIMER INTERRUPTED IN LESS THAN 12 SECONDS, SO NOW CHECK TO SEE IF IT
                                ;HAPPENED IN LESS THAN 8.
                                ;
032652' 023727  000332' 000764  40$:    CMP     METER,#500.         ;DID INTERRUPT OCCUR BEFORE 8 SECS.?
032660' 003416                          BLE     60$                 ;NO, OK
032662' 012700  001161                  MOV     #625.,R0
032666' 163700  000332'                 SUB     METER,R0            ;CALC HOW MUCH TIME
```

```
032672' 005001                       CLR     R1                      ;CONTAINS SECS
032674' 162700  000077       50$:    SUB     #63.,R0                 ;FIND OUT HOW MAY SECS IT TOOK
032700' 103402                       BCS     55$
032702' 005201                       INC     R1
032704' 000773                       BR      50$
032706'                      55$:    ERRHRD  070,TIMTST,MSG11        ;PRINT ERROR MESSAGE
032706' 104456                                                                               TRAP    C$ERHRD
032710' 000106                                                                               .WORD   70
032712' 002216'                                                                              .WORD   TIMTST
032714' 013402'                                                                              .WORD   MSG11
032716' 004737  017362'      60$:    JSR     PC,CLRDNI               ;CLEAR DNI
032722' 103004                       BCC     70$
032724'                              ERRHRD  071,TIMTST,RACMG7       ;ERROR DNI DID NOT CLEAR
032724' 104456                                                                               TRAP    C$ERHRD
032726' 000107                                                                               .WORD   71
032730' 002216'                                                                              .WORD   TIMTST
032732' 012670'                                                                              .WORD   RACMG7
032734'                      70$:
032734'                        ENDSEG
032734'                                                                              10001$:
032734' 104405                                                                               TRAP    C$ESEG
032736'                      ENDTST
032736'                                                                              L10132:
032736' 104401                                                                               TRAP    C$ETST
```

```
                                        .SBTTL  TEST 17: LINK MEMORY TEST

                                        ;*******************************************************************
                                        ;
                                        ;THIS TEST WILL EXHAUSTIVELY TEST THE LINK MEMORY.
                                        ;
                                        ;THE LINK MEMORY OCCUPIES THE 16-32K ADDRESS SPACE OF THE T-11.
                                        ;CUSTOM MICROCODE MODULE C MICROTEST #3 IS USED TO DO THE ACTUAL TESTING.
                                        ;MICROTEST #3 RUNS A SERIES OF MICROSUBTESTS TESTS ON THE LINK MEMORY CHECKING
                                        ;FOR BOTH ADDRESS AND DATA ERRORS. IF AN ERROR DOES OCCUR THE PORT CONTROL
                                        ;BLOCK WILL CONTAIN THE INFORMATION ABOUT THE ERROR.
                                        ;
                                        ;PCBB+0: CONTAINS THE MICROSUBTEST THAT FAILED
                                        ;PCBB+1: 0 = DATA ERROR, 1 = ADDRESS ERROR,
                                        ;PCBB+2: CONTAINS THE ADDRESS OF THE LOCATION
                                        ;PCBB+4: CONTAINS THE DATA THAT WAS WRITTEN
                                        ;PCBB+6: CONTAINS THE DATA THAT WAS READ
                                        ;
                                        ;MICROSUBTEST #         DESCRIPTION
                                        ;
                                        ;       1               ACCESS TEST
                                        ;       2               ADDRESS SHIFT TEST
                                        ;       3               DATA LATCH TEST
                                        ;       4               ADDRESS BIT SHIFT #1
                                        ;       5               ADDRESS BIT SHIFT #2
                                        ;       6               MARCH TEST
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROMODULE 'C' INTO THE TOP HALF OF WCS IF NOT ALREADY DONE SO
                                        ;       2-WAIT FOR THE MICROMONITOR TO BECOME ACTIVE
                                        ;       3-TELL MICROMONITOR TO EXECUTE MICROTEST #3
                                        ;       4-VERIFY DNI SET INDICATING TEST COMPLETE
                                        ;       5-CHECK STATE FIELD OF PCSR1 FOR ERROR CONDITION
                                        ;
                                        ;*******************************************************************

032740'                                 BGNTST
032740'                                                                         T17::
                                        ;
                                        ;CHECK TO SEE IF MODULE 'C' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
032740'                                  BGNSEG
032740' 104404                                                                          TRAP    C$BSEG
032742' 022737  000103  000326'         CMP     #'C,MICRO               ;HAS MICROCODE MODULE 'C' BEEN LOADED?
032750' 001004                          BNE     5$                      ;NO
032752' 122777  000001  145360          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
032760' 001440                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
032762' 012737  000103  000326' 5$:     MOV     #'C,MICRO               ;GO LOAD MICROCODE MODULE C
032770' 004737  020340'                 JSR     PC,LODMIC
032774' 103002                          BCC     10$                     ;OK
032776'                                 ESCAPE  TST
032776' 104410                                                                          TRAP    C$ESCAPE
033000' 000404                                                                          .WORD   L10133-.
033002' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
```

```
033010' 004737  017316'                  JSR     PC,CHKDNI
033014' 103022                           BCC     20$                        ;OK
033016' 012737  001000' 000310'          MOV     #SDNI,BITNAM
033024' 012737  001277' 000312'          MOV     #SNSET,BITSTA
033032' 012737  001342' 000314'          MOV     #SAFTER,PWHEN
033040' 012737  001357' 000316'          MOV     #SGTCMD,PCOMND
033046'                                  ERRHRD  072,LNKMEM,MSG1 ;PRINT ERROR MESSAGE
033046' 104456                                                                        TRAP    C$ERHRD
033050' 000110                                                                        .WORD   72
033052' 002240'                                                                       .WORD   LNKMEM
033054' 012716'                                                                       .WORD   MSG1
033056'                                  ESCAPE TST
033056' 104410                                                                        TRAP    C$ESCAPE
033060' 000324                                                                        .WORD   L10133-.
033062' 004737  017362'          20$:    JSR     PC,CLRDNI                  ;CLEAR DNI
033066' 103006                           BCC     25$
033070'                                  ERRHRD  073,LNKMEM,RACMG7
033070' 104456                                                                        TRAP    C$ERHRD
033072' 000111                                                                        .WORD   73
033074' 002240'                                                                       .WORD   LNKMEM
033076' 012670'                                                                       .WORD   RACMG7
033100'                                  ESCAPE TST
033100' 104410                                                                        TRAP    C$ESCAPE
033102' 000302                                                                        .WORD   L10133-.
033104'                          25$:
033104'                            ENDSEG
033104'                                                                     10000$:
033104' 104405                                                                        TRAP    C$ESEG

                                 ;
                                 ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE, CLEAR THE
                                 ;LOCATIONS FOR ERROR INFORMATION, LOAD COMMAND FIELD OF PCSR0 WITH A 3
                                 ;CAUSING THE MICROMONITOR TO EXECUTE MICROTEST #3, THIS WILL START THE
                                 ;EXECUTION OF THE MICROSUBTEST SEQUENCE OF MEMORY TESTS. DNI WILL SET
                                 ;WHEN THE TEST IS COMPLETE
                                 ;
033106'                            BGNSEG
033106' 104404                                                                        TRAP    C$BSEG
033110' 004737  020060'                  JSR     PC,CHKMON                  ;WAIT FOR MICROMONITOR
033114' 103006                           BCC     30$                        ;OK
033116'                                  ERRHRD  074,LNKMEM,MSG46           ;PRINT ERROR
033116' 104456                                                                        TRAP    C$ERHRD
033120' 000112                                                                        .WORD   74
033122' 002240'                                                                       .WORD   LNKMEM
033124' 016666'                                                                       .WORD   MSG46
033126'                                  ESCAPE  TST                        ;LEAVE TEST
033126' 104410                                                                        TRAP    C$ESCAPE
033130' 000254                                                                        .WORD   L10133-.
033132' 012777  000606' 145202  30$:     MOV     #PCBB,@PCSR2               ;TELL MICROCODE TEST WHERE PCBB IS
033140' 005077  145200                   CLR     @PCSR3
033144' 005037  000606'                  CLR     PCBB+0                     ;CLEAR OUT THE PCBB
033150' 005037  000610'                  CLR     PCBB+2
033154' 005037  000612'                  CLR     PCBB+4
033160' 005037  000614'                  CLR     PCBB+6
033164' 012777  000003  145144           MOV     #3,@PCSR0                  ;TELL MICROMONITOR TO EXECUTE...
                                                                            ;MICROTEST #3
033172' 012737  000770  000332'          MOV     #10*SECOND,METER           ;PUT SOME TIME ON THE METER
```

```
033200' 004737  017316'                 JSR     PC,CHKDNI               ;WAIT FOR MICROTEST TO FINISH
033204' 103021                          BCC     40$                     ;OK, IT FINISHED
033206' 004737  020132'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
033212' 103006                          BCC     35$                     ;NO, OK
033214'                                 ERRHRD  075,LNKMEM,MSG44        ;PRINT ERROR MESSAGE
033214' 104456                                                                                  TRAP    C$ERHRD
033216' 000113                                                                                  .WORD   75
033220' 002240'                                                                                 .WORD   LNKMEM
033222' 016442'                                                                                 .WORD   MSG44
033224'                                 ESCAPE  TST                     ;LEAVE TEST
033224' 104410                                                                                  TRAP    C$ESCAPE
033226' 000156                                                                                  .WORD   L10133-.
033230' 012702  000003          35$:    MOV     #3,R2                   ;MICROTEST #
033234'                                 ERRHRD  076,LNKMEM,MSG12        ;TELL MICROTEST HUNG
033234' 104456                                                                                  TRAP    C$ERHRD
033236' 000114                                                                                  .WORD   76
033240' 002240'                                                                                 .WORD   LNKMEM
033242' 013466'                                                                                 .WORD   MSG12
033244'                                 ESCAPE  TST
033244' 104410                                                                                  TRAP    C$ESCAPE
033246' 000136                                                                                  .WORD   L10133-.
                                ;
                                ;MICROTEST IS COMPLETE, NOW CHECK FOR AN ERROR CONDITION
                                ;
033250' 122777  000003  145062  40$:    CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
033256' 001027                          BNE     47$                     ;NO
033260' 122737  000000  000607'         CMPB    #DATERR,PCBB+1          ;YES, WAS IT A DATA ERROR?
033266' 001003                          BNE     45$                     ;NO
033270' 012702  001446'                 MOV     #SDATER,R2              ;YES, POINT TO DATA ERROR STRING
033274' 000406                          BR      46$
033276' 122737  000001  000607' 45$:    CMPB    #ADRERR,PCBB+1          ;WAS IT AN ADDRESS ERROR
033304' 001014                          BNE     47$                     ;NO
033306' 012702  001461'                 MOV     #SADRER,R2              ;POINT TO ADDRESS ERROR STRING
033312' 013701  000610'         46$:    MOV     PCBB+2,R1               ;GET FAILING ADDRESS
033316' 013703  000612'                 MOV     PCBB+4,R3               ;GET GOOD DATA
033322' 013704  000614'                 MOV     PCBB+6,R4               ;GET BAD DATA
033326'                                 ERRHRD  077,LNKMEM,MSG16        ;PRINT ERROR MESSAGE
033326' 104456                                                                                  TRAP    C$ERHRD
033330' 000115                                                                                  .WORD   77
033332' 002240'                                                                                 .WORD   LNKMEM
033334' 013636'                                                                                 .WORD   MSG16
033336' 032777  010000  144772  47$:    BIT     #PARERR,@PCSR0          ;DID A PARITY ERROR OCCUR?
033344' 001407                          BEQ     50$                     ;NO
033346'                                 ERRHRD  078,LNKMEM,MSG43        ;YES PRINT ERROR
033346' 104456                                                                                  TRAP    C$ERHRD
033350' 000116                                                                                  .WORD   78
033352' 002240'                                                                                 .WORD   LNKMEM
033354' 016420'                                                                                 .WORD   MSG43
033356' 012777  010000  144752          MOV     #PARERR,@PCSR0          ;CLEAR PARITY ERROR FLAG

                                ;
                                ;WRITE ONE TO CLEAR THE DNI BIT
                                ;
033364' 004737  017362'         50$:    JSR     PC,CLRDNI               ;CLEAR DNI
033370' 103004                          BCC     55$
033372'                                 ERRHRD  079,LNKMEM,RACMG7
```

```
033372' 104456                                          TRAP    C$ERHRD
033374' 000117                                          .WORD   79
033376' 002240'                                         .WORD   LNKMEM
033400' 012670'                                         .WORD   RACMG7
033402'                 55$:
033402'                   ENDSEG
033402'                                        10001$:
033402' 104405                                          TRAP    C$ESEG
033404'                 ENDTST
033404'                                        L10133:
033404' 104401                                          TRAP    C$ETST
```

```
                                       .SBTTL  TEST 18: DMA 'TO' ADDRESS TEST

                                       ;*********************************************************************
                                       ;
                                       ;THIS TEST WILL VERIFY THAT THE INTERNAL REGISTER 'DMATO' CAN BE READ
                                       ;AND WRITTEN. THE T11 WILL BE USED TO WRITE AND READ THIS REGISTER.
                                       ;THIS TEST REQUIRES THE USE OF CUSTOM MICROCODE MODULE C MICROTEST #4.
                                       ;PCBB+0 WILL BE WRITTEN WITH THE DATA PATTERN TO TEST, THE T11 WILL
                                       ;WRITE THIS PATTERN TO THE 'DMATO' REGISTER AND READ IT BACK AND PUT
                                       ;THE DATA READ INTO PCBB+2. THE DATA AT PCBB+2 WILL BE VERIFIED.
                                       ;
                                       ;TEST SEQUENCE:
                                       ;        1-LOAD MICROCODE MODULE 'C' IF NOT ALREADY DONE SO
                                       ;        2-LOAD PCBB+0 WITH DATA PATTERN
                                       ;        3-START MICROCODE
                                       ;        4-WAIT FOR 'DNI'
                                       ;        5-VERIFY PCBB+2 FOR CORRECT PATTERN
                                       ;        6-REPEAT STEPS 2-6 FOR ALL DATA PATTERNS
                                       ;
                                       ;*********************************************************************

033406'                                BGNTST
033406'                                                                                       T18::
                                       ;
                                       ;CHECK TO SEE IF MODULE 'C' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                       ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                       ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                       ;
033406'                                  BGNSEG
033406' 104404                                                                                     TRAP    C$BSEG
033410' 022737  000103  000326'                 CMP     #'C,MICRO       ;HAS MICROCODE MODULE 'C' BEEN LOADED?
033416' 001004                                  BNE     5$                      ;NO
033420' 122777  000001  144712                  CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
033426' 001440                                  BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
033430' 012737  000103  000326' 5$:             MOV     #'C,MICRO       ;NO
033436' 004737  020340'                         JSR     PC,LODMIC       ;GO LOAD MICROCODE MODULE C
033442' 103002                                  BCC     10$             ;OK
033444'                                         ESCAPE TST
033444' 104410                                                                                     TRAP    C$ESCAPE
033446' 000314                                                                                     .WORD   L10134-.
033450' 012737  000176  000332' 10$:            MOV     #2*SECOND,METER
033456' 004737  017316'                         JSR     PC,CHKDNI       ;WAIT FOR MICROMONITOR TO TAKE OVER
033462' 103022                                  BCC     20$             ;OK
033464' 012737  001000' 000310'                 MOV     #$DNI,BITNAM
033472' 012737  001277' 000312'                 MOV     #$NSET,BITSTA
033500' 012737  001342' 000314'                 MOV     #$AFTER,PWHEN
033506' 012737  001357' 000316'                 MOV     #$GTCMD,PCOMND
033514'                                         ERRHRD  080,DMATO,MSG1
033514' 104456                                                                                     TRAP    C$ERHRD
033516' 000120                                                                                     .WORD   80
033520' 002270'                                                                                    .WORD   DMATO
033522' 012716'                                                                                    .WORD   MSG1
033524'                                         ESCAPE TST
033524' 104410                                                                                     TRAP    C$ESCAPE
033526' 000234                                                                                     .WORD   L10134-.
033530' 004737  017362'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI
```

```
033534' 103006                              BCC     25$
033536'                                     ERRHRD  081,DMATO,RACMG7
033536' 104456                                                                      TRAP    C$ERHRD
033540' 000121                                                                      .WORD   81
033542' 002270'                                                                     .WORD   DMATO
033544' 012670'                                                                     .WORD   RACMG7
033546'                                     ESCAPE TST
033546' 104410                                                                      TRAP    C$ESCAPE
033550' 000212                                                                      .WORD   L10134-.
033552'                             25$:
033552'                               ENDSEG
033552'                                                                     10000$:
033552' 104405                                                                      TRAP    C$ESEG

                                    ;
                                    ;POINT TO LIST OF DATA PATTERNS TO USE, THERE ARE FIVE ENTRIES IN THE LIST
                                    ;SO THE LOOP WILL BE EXECUTED 5 TIMES ONCE FOR EACH DATA PATTERN.
                                    ;
033554' 012701  000520'                     MOV     #PATERN,R1      ;GET ADDRESS OF DATA PATTERNS
033560' 012705  000005                      MOV     #5,R5           ;NUMBER OF DATA PATTERNS
033564'                             30$:
                                    ;
                                    ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE, LOAD PCBB+0 WITH
                                    ;A DATA PATTERN, LOAD THE COMMAND FIELD OF PCSRO WITH 4 TO START THE EXECUTION
                                    ;OF MICROTEST #4, DNI SETS WHEN IT IS COMPLETE
                                    ;
033564'                               BGNSEG
033564' 104404                                                                      TRAP    C$BSEG
033566' 004737  020060'                     JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
033572' 103006                              BCC     35$                     ;OK
033574'                                     ERRHRD  082,DMATO,MSG46         ;PRINT ERROR
033574' 104456                                                                      TRAP    C$ERHRD
033576' 000122                                                                      .WORD   82
033600' 002270'                                                                     .WORD   DMATO
033602' 016666'                                                                     .WORD   MSG46
033604'                                     ESCAPE  TST                     ;LEAVE TEST
033604' 104410                                                                      TRAP    C$ESCAPE
033606' 000154                                                                      .WORD   L10134-.
033610' 012137  000606'             35$:    MOV     (R1)+,PCBB+0    ;GET A DATA PATTERN
033614' 012777  000004  144514              MOV     #4,@PCSRO       ;START MICROTEST #4
033622' 012737  000176  000332'             MOV     #2*SECOND,METER ;PUT SOME TIME ON THE METER
033630' 004737  017316'                     JSR     PC,CHKDNI       ;WAIT FOR DNI
033634' 103021                              BCC     40$             ;OK
033636' 004737  020132'                     JSR     PC,CHKINT       ;SEE IF ANY ERROR INTERRUPTS OCCURRED
033642' 103006                              BCC     36$             ;NO, OK
033644'                                     ERRHRD  083,DMATO,MSG44 ;PRINT ERROR MESSAGE
033644' 104456                                                                      TRAP    C$ERHRD
033646' 000123                                                                      .WORD   83
033650' 002270'                                                                     .WORD   DMATO
033652' 016442'                                                                     .WORD   MSG44
033654'                                     ESCAPE  TST             ;LEAVE TEST
033654' 104410                                                                      TRAP    C$ESCAPE
033656' 000104                                                                      .WORD   L10134-.
033660' 012702  000004              36$:    MOV     #4,R2           ;MICROTEST #
033664'                                     ERRHRD  084,DMATO,MSG12 ;TELL MICROTEST HUNG
033664' 104456                                                                      TRAP    C$ERHRD
033666' 000124                                                                      .WORD   84
```

```
033670' 002270'                                                                        .WORD   DMATO
033672' 013466'                                                                        .WORD   MSG12
033674'                                 ESCAPE  TST
033674' 104410                                                                         TRAP    C$ESCAPE
033676' 000064                                                                         .WORD   L10134-.
                                ;
                                ;OK NOW CHECK TO SEE IF DATA READ IS SAME AS THE DATA WRITTEN
                                ;REMEMBER BIT 0 OF DMATO IS NOT USED
                                ;
033700' 013703  000606'         40$:    MOV     PCBB+0,R3          ;GET ORIGINAL DATA PATTERN
033704' 042703  000001                  BIC     #BIT0,R3           ;STRIP LSB
033710' 020337  000610'                 CMP     R3,PCBB+2          ;SEE IF DATA WRITTEN = DATA READ
033714' 001407                          BEQ     50$                ;YES
033716' 013704  000610'                 MOV     PCBB+2,R4          ;NO, ERROR
033722'                                 ERRHRD  085,DMATO,MSG13 ;PRINT ERROR
033722' 104456                                                                         TRAP    C$ERHRD
033724' 000125                                                                         .WORD   85
033726' 002270'                                                                        .WORD   DMATO
033730' 013512'                                                                        .WORD   MSG13
033732'                          ENDSEG
033732'                                                                        10001$:
033732' 104405                                                                         TRAP    C$ESEG
                                ;
                                ;WRITE ONE TO CLEAR DNI BIT
                                ;
033734' 004737  017362'         50$:    JSR     PC,CLRDNI          ;CLEAR DNI
033740' 103006                          BCC     55$
033742'                                 ERRHRD  086,DMATO,RACMG7;ERROR DNI DID NOT CLEAR
033742' 104456                                                                         TRAP    C$ERHRD
033744' 000126                                                                         .WORD   86
033746' 002270'                                                                        .WORD   DMATO
033750' 012670'                                                                        .WORD   RACMG7
033752'                                 ESCAPE  TST
033752' 104410                                                                         TRAP    C$ESCAPE
033754' 000006                                                                         .WORD   L10134-.
033756' 005305                  55$:    DEC     R5                 ;ANY MORE DATA PATTERNS?
033760' 001301                          BNE     30$                ;YES
033762'                         ENDTST
033762'                                                                        L10134:
033762' 104401                                                                         TRAP    C$ETST
```

```
                                        .SBTTL  TEST 19: DMA 'FROM' ADDRESS REGISTER TEST

                                        ;*******************************************************************
                                        ;
                                        ;THIS TEST CHECKS THE OPERATION OF THE REGISTER/COUNTER THAT CONTAINS
                                        ;THE ADDRESS OF THE LINK MEMORY WORD TO BE MOVED TO THE HOST DURING
                                        ;DMA OPERATIONS. THE REGISTER CAN BE WRITTEN BY THE T11 BUT IT CAN
                                        ;NOT BE READ BACK FOR VERIFICATION, THEREFORE IT MUST BE CHECKED
                                        ;INDIRECTLY.
                                        ;
                                        ;THE METHOD USED IS TO LOAD MICROCODE MODULE C IF IT HAS NOT ALREADY
                                        ;BEEN DONE. THE MICROTEST #5 LOADS EACH LOCATION OF LINK MEMORY WITH
                                        ;ITS ADDRESS THEN IT TAKES THE CONTENTS OF PCBB+0 AND LOADS IT INTO
                                        ;THE DMA 'FROM' ADDRESS REGISTER, THE 'TO' REGISTER IS LOADED WITH
                                        ;THE ADDRESS OF PCBB+2, THE WORD COUNT IS LOADED FOR A ONE WORD TRANSFER
                                        ;AND THE DMA ENGINE IS STARTED. THE HOST VERIFIES PCBB+2 = PCBB+0
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROCODE MODULE 'C' IF NOT ALREADY DONE SO
                                        ;       2-LOAD PCBB+0 WITH ADDRESS OF LINK MEMORY TO LOAD
                                        ;       3-START MICROCODE
                                        ;       4-WAIT FOR 'DNI'
                                        ;       5-VERIFY PCBB+2 = PCBB+0
                                        ;       6-CHANGE PCBB+0 BY 1K CHUNKS
                                        ;       7-REPEAT 2-6
                                        ;
                                        ;*************************************************************************

 033764'                                BGNTST
 033764'                                                                        T19::
                                        ;
                                        ;CHECK TO SEE IF MODULE 'C' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
 033764'                                 BGNSEG
 033764' 104404                                                                         TRAP    C$BSEG
 033766' 022737  000103  000326'                CMP     #'C,MICRO               ;HAS MICRO MODULE C BEEN LOADED?
 033774' 001004                                 BNE     5$                      ;NO
 033776' 122777  000001  144334                 CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
 034004' 001440                                 BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
 034006' 012737  000103  000326' 5$:            MOV     #'C,MICRO               ;NO, LOAD MICRO MODULE C
 034014' 004737  020340'                        JSR     PC,LODMIC
 034020' 103002                                 BCC     10$                     ;OK
 034022'                                        ESCAPE TST
 034022' 104410                                                                         TRAP    C$ESCAPE
 034024' 000316                                                                         .WORD   L10135-.
 034026' 012737  000176  000332' 10$:           MOV     #2*SECOND,METER         ;PUT SOME TIME ON THE METER
 034034' 004737  017316'                        JSR     PC,CHKDNI               ;WAIT FOR THE MICROMONITOR
 034040' 103022                                 BCC     20$                     ;OK
 034042' 012737  001000' 000310'                MOV     #$DNI,BITNAM
 034050' 012737  001277' 000312'                MOV     #$NSET,BITSTA
 034056' 012737  001342' 000314'                MOV     #$AFTER,PWHEN
 034064' 012737  001357' 000316'                MOV     #$GTCMD,PCOMND
 034072'                                        ERRHRD  087,DMAFRM,MSG1
 034072' 104456                                                                         TRAP    C$ERHRD
```

```
034074' 000127                                                                  .WORD   87
034076' 002336'                                                                 .WORD   DMAFRM
034100' 012716'                                                                 .WORD   MSG1
034102'                                 ESCAPE TST
034102' 104410                                                                  TRAP    C$ESCAPE
034104' 000236                                                                  .WORD   L10135-.
034106' 004737  017362'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
034112' 103006                          BCC     25$
034114'                                 ERRHRD  088,DMAFRM,RACMG7       ;DNI BIT DID NOT CLEAR!
034114' 104456                                                                  TRAP    C$ERHRD
034116' 000130                                                                  .WORD   88
034120' 002336'                                                                 .WORD   DMAFRM
034122' 012670'                                                                 .WORD   RACMG7
034124'                                 ESCAPE  TST
034124' 104410                                                                  TRAP    C$ESCAPE
034126' 000214                                                                  .WORD   L10135-.
034130'                         25$:
034130'                           ENDSEG
034130'                                                                 10000$:
034130' 104405                                                                  TRAP    C$ESEG
034132' 005037  000606'                 CLR     PCBB+0                  ;TELL T11 TO START LOAD AT BASE
                                                                        ;OF LINK MEMORY
034136'                         30$:
                                ;
                                ;WAIT FOR THE MICRMONITOR TO ENTER 'IN MONITOR' STATE, LOAD COMMAND FIELD
                                ;OF PCSR0 WITH 5 TO START THE EXECUTION OF MICROTEST #5, WAIT FOR DNI
                                ;
034136'                           BGNSEG
034136' 104404                                                                  TRAP    C$BSEG
034140' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
034144' 103006                          BCC     40$                     ;OK
034146'                                 ERRHRD  089,DMAFRM,MSG46        ;PRINT ERROR
034146' 104456                                                                  TRAP    C$ERHRD
034150' 000131                                                                  .WORD   89
034152' 002336'                                                                 .WORD   DMAFRM
034154' 016666'                                                                 .WORD   MSG46
034156'                                 ESCAPE  TST                     ;LEAVE TEST
034156' 104410                                                                  TRAP    C$ESCAPE
034160' 000162                                                                  .WORD   L10135-.
034162' 012777  000005  144146  40$:    MOV     #5,@PCSR0               ;TELL T11 TO START MICROTEST #5
034170' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT A WHILE FOR DNI
034176' 004737  017316'                 JSR     PC,CHKDNI
034202' 103021                          BCC     50$
034204' 004737  020132'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
034210' 103006                          BCC     45$                     ;NO, OK
034212'                                 ERRHRD  090,DMAFRM,MSG44        ;PRINT ERROR MESSAGE
034212' 104456                                                                  TRAP    C$ERHRD
034214' 000132                                                                  .WORD   90
034216' 002336'                                                                 .WORD   DMAFRM
034220' 016442'                                                                 .WORD   MSG44
034222'                                 ESCAPE  TST                     ;LEAVE TEST
034222' 104410                                                                  TRAP    C$ESCAPE
034224' 000116                                                                  .WORD   L10135-.
034226' 012702  000005          45$:    MOV     #5,R2                   ;WE WERE EXECUTING TEST 5
034232'                                 ERRHRD  091,DMAFRM,MSG12        ;TELL HIM IT HUNG
034232' 104456                                                                  TRAP    C$ERHRD
```

```
034234' 000133                                                                   .WORD   91
034236' 002336'                                                                  .WORD   DMAFRM
034240' 013466'                                                                  .WORD   MSG12
034242'                                   ESCAPE TST
034242' 104410                                                                   TRAP    C$ESCAPE
034244' 000076                                                                   .WORD   L10135-.
                          ;
                          ;OK GET THE 'FROM ADDRESS' THAT I PASSED TO THE MICROCODE AND MAKE IT AN
                          ;ACTUAL LINK MEMORY ADDRESS BY FORCING BIT 15 THIS MAPS IT TO THE 16-32K RANGE
                          ;ALSO FORCE BIT 2 BECAUSE THE FIRST 2 WORDS OF EACH BUFFER CAN NOT BE DMA'ED
                          ;FROM LINK MEMORY TO THE UNIBUS. CHECK THAT THE DATA TRANSFERRED IS GOOD
                          ;
034246' 013703 000606'    50$:    MOV     PCBB+0,R3               ;GET ORIGINAL 'FROM' ADDRESS
034252' 052703 100004             BIS     #BIT15!BIT2,R3          ;MAKE IT ACTUAL LINK MEMORY ADDRESS
034256' 013704 000610'            MOV     PCBB+2,R4               ;GET WHAT WAS READ FROM LINK MEMORY
034262' 020304                    CMP     R3,R4                   ;IS DATA CORRECT?
034264' 001404                    BEQ     55$                     ;YES
034266'                           ERRHRD  092,DMAFRM,MSG14        ;NO
034266' 104456                                                                   TRAP    C$ERHRD
034270' 000134                                                                   .WORD   92
034272' 002336'                                                                  .WORD   DMAFRM
034274' 013560'                                                                  .WORD   MSG14
034276'                   55$:
034276'                    ENDSEG
034276'                                                              10001$:
034276' 104405                                                                   TRAP    C$ESEG
                          ;
                          ;WRITE ONE TO CLEAR DNI
                          ;
034300' 004737 017362'            JSR     PC,CLRDNI
034304' 103006                    BCC     57$
034306'                           ERRHRD  093,DMAFRM,RACMG7       ;ERROR DNI DID NOT CLEAR!
034306' 104456                                                                   TRAP    C$ERHRD
034310' 000135                                                                   .WORD   93
034312' 002336'                                                                  .WORD   DMAFRM
034314' 012670'                                                                  .WORD   RACMG7
034316'                           ESCAPE  TST
034316' 104410                                                                   TRAP    C$ESCAPE
034320' 000022                                                                   .WORD   L10135-.
034322' 022737 074000 000606' 57$: CMP    #74000,PCBB+0           ;HAVE WE CHECKED ALL 1K CHUNKS?
034330' 001404                    BEQ     60$                     ;YES
034332' 062737 004000 000606'     ADD     #4000,PCBB+0            ;NO, CHECK NEXT 1K
034340' 000676                    BR      30$                     ;DO AGAIN
034342'                   60$:
034342'                   ENDTST
034342'                                                              L10135:
034342' 104401                                                                   TRAP    C$ETST
```

```
                                        .SBTTL  TEST 20: DMA BLOCK TRANSFER TEST

                                        ;*************************************************************************
                                        ;
                                        ;THIS TEST WILL VERIFY THAT THE DMA ENGINE CAN TRANSFER A MAXIMUM SIZE DATA
                                        ;BLOCK TO HOST MEMORY.
                                        ;
                                        ;THIS TEST USES CUSTOM MICROCODE MODULE C, MICROTEST #6. THE MICROTEST
                                        ;FILLS EACH LOCATION OF LINK MEMORY WITH ITS ADDRESS AND THEN SETS
                                        ;UP A TRANSFER FROM LINK MEMORY TO THE ADDRESS POINTED TO BY PCBB+0.
                                        ;THE TRANSFER SIZE IS 1776 WORDS. AFTER THE MICROTEST FINISHES THE
                                        ;BUFFER IS CHECKED TO SEE IF IT CONTAINS THE INCREMENTING ADDRESS PATTERN.
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROCODE MODULE 'C' IF NOT ALREADY DONE SO
                                        ;       2-LOAD PCBB+0 WITH BUFFER ADDRESS
                                        ;       3-START MICROCODE
                                        ;       4-WAIT FOR DNI
                                        ;       5-VERIFY ALL 1776 WORDS STARTING AT BUFFER ADDRESS BASE
                                        ;
                                        ;*************************************************************************

 034344'                                BGNTST
 034344'                                                                                T20::
                                        ;
                                        ;CHECK TO SEE IF MODULE 'C' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                        ;THE TOP HALF OF WCS. START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
 034344'                                 BGNSEG
 034344' 104404                                                                            TRAP    C$BSEG
 034346' 022737  000103  000326'             CMP     #'C,MICRO          ;HAS MICRO MODULE C BEEN LOADED?
 034354' 001004                              BNE     5$                 ;NO
 034356' 122777  000001  143754              CMPB    #INMON,@PCSR1      ;YES, IS THE MICROMONITOR ACTIVE?
 034364' 001435                              BEQ     20$                ;YES SKIP LOADING THE MICROMODULE
 034366' 012737  000103  000326' 5$:         MOV     #'C,MICRO          ;NO, LOAD MICRO MODULE C
 034374' 004737  020340'                     JSR     PC,LODMIC
 034400' 103530                              BCS     70$                ;ERROR
 034402' 012737  000176  000332' 10$:        MOV     #2*SECOND,METER    ;PUT SOME TIME ON THE METER
 034410' 004737  017316'                     JSR     PC,CHKDNI          ;WAIT FOR THE MICROMONITOR
 034414' 103021                              BCC     20$                ;OK
 034416' 012737  001000' 000310'             MOV     #$DNI,BITNAM
 034424' 012737  001277' 000312'             MOV     #$NSET,BITSTA
 034432' 012737  001342' 000314'             MOV     #$AFTER,PWHEN
 034440' 012737  001357' 000316'             MOV     #$GTCMD,PCOMND
 034446'                                     ERRHRD  094,DMABLK,MSG1
 034446' 104456                                                                            TRAP    C$ERHRD
 034450' 000136                                                                            .WORD   94
 034452' 002406'                                                                           .WORD   DMABLK
 034454' 012716'                                                                           .WORD   MSG1
 034456' 000501                              BR      70$
 034460' 004737  017362'         20$:        JSR     PC,CLRDNI          ;GO CLEAR THE DNI BIT
 034464' 103005                              BCC     25$
 034466'                                     ERRHRD  095,DMABLK,RACMG7  ;DNI DID NOT CLEAR!
 034466' 104456                                                                            TRAP    C$ERHRD
 034470' 000137                                                                            .WORD   95
```

```
034472' 002406'                                                            .WORD   DMABLK
034474' 012670'                                                            .WORD   RACMG7
034476' 000471                            BR      70$
034500'                           25$:
034500'                            ENDSEG
034500'                                                             10000$:
034500' 104405                                                             TRAP    C$ESEG

                                  ;
                                  ;TELL MICROCODE TO DMA TO A BUFFER IN FREE MEMORY LOCATED ABOVE THIS DIAGNOSTIC
                                  ;BY LOADING PCBB+0 WITH THE ADDRESS TO DMA TO. WAIT FOR THE MICROMONITOR TO
                                  ;ENTER THE 'IN MONITOR' STATE AND LOAD THE COMMAND FIELD OF PCSR0 WITH A 6
                                  ;TO START THE EXECUTION OF MICROTEST #6. WAIT FOR DNI TO SET INDICATING IT IS
                                  ;COMPLETE
                                  ;
034502'                            BGNSEG
034502' 104404                                                             TRAP    C$BSEG
034504' 013737  000324' 000606'           MOV     FREMEM,PCBB+0         ;GET ADDRESS OF FREE MEMORY
034512' 004737  020060'                   JSR     PC,CHKMON             ;WAIT FOR MICROMONITOR
034516' 103006                            BCC     30$                   ;OK
034520'                                   ERRHRD  096,DMABLK,MSG46      ;PRINT ERROR
034520' 104456                                                             TRAP    C$ERHRD
034522' 000140                                                             .WORD   96
034524' 002406'                                                            .WORD   DMABLK
034526' 016666'                                                            .WORD   MSG46
034530'                                   ESCAPE  TST                   ;LEAVE TEST
034530' 104410                                                             TRAP    C$ESCAPE
034532' 000136                                                             .WORD   L10136-.
034534' 012777  000006  143574  30$:      MOV     #6,@PCSR0             ;START MICROTEST #6
034542' 012737  000473  000332'           MOV     #5*SECOND,METER       ;PUT SOME TIME ON THE METER
034550' 004737  017316'                   JSR     PC,CHKDNI             ;WAIT FOR MICROTEST FINISH
034554' 103020                            BCC     40$                   ;OK
034556' 004737  020132'                   JSR     PC,CHKINT             ;SEE IF ANY ERROR INTERRUPTS OCCURRED
034562' 103006                            BCC     35$                   ;NO, OK
034564'                                   ERRHRD  097,DMABLK,MSG44      ;PRINT ERROR MESSAGE
034564' 104456                                                             TRAP    C$ERHRD
034566' 000141                                                             .WORD   97
034570' 002406'                                                            .WORD   DMABLK
034572' 016442'                                                            .WORD   MSG44
034574'                                   ESCAPE  TST                   ;LEAVE TEST
034574' 104410                                                             TRAP    C$ESCAPE
034576' 000072                                                             .WORD   L10136-.
034600' 012702  000006          35$:      MOV     #6,R2                 ;MICROTEST NEVER FINISHED!
034604'                                   ERRHRD  098,DMABLK,MSG12
034604' 104456                                                             TRAP    C$ERHRD
034606' 000142                                                             .WORD   98
034610' 002406'                                                            .WORD   DMABLK
034612' 013466'                                                            .WORD   MSG12
034614' 000422                            BR      70$
                                  ;
                                  ;OK NOW CHECK THE DATA TRANSFERRED FROM LINK MEMORY TO UNIBUS MEMORY.
                                  ;THE DATA PATTERN IS ACTUALLY THE ADDRESS FROM WHICH THE DATA CAME FROM.
                                  ;THE FIRST LOCATION WILL BE 4 BYTES FROM THE BEGINING OF LINK MEMORY BECAUSE
                                  ;THE DMA FROM ADDRESS REGISTER DOES NOT USE BITS 0 AND 1.
                                  ;
034616' 013702  000606'         40$:      MOV     PCBB+0,R2             ;GET ADDRESS OF BUFFER
034622' 012746  100004                    MOV     #100004,-(SP)         ;STARTING DATA PATTERN
```

```
7987 034626' 021216          50$:    CMP     (R2),(SP)               ;IS DATA PATTERN CORRECT?
7988 034630' 001405                  BEQ     60$                     ;YES
7989 034632' 010601                  MOV     SP,R1                   ;NO GET GOOD DATA
7990 034634'                         ERRHRD  099,DMABLK,MSG15        ;PRINT ERROR MESSAGE
7991 034634' 104456                                                          TRAP    C$ERHRD
7992 034636' 000143                                                          .WORD   99
7993 034640' 002406'                                                         .WORD   DMABLK
7994 034642' 013606'                                                         .WORD   MSG15
7995 034644' 021627 103776   60$:    CMP     (SP),#103776            ;DONE ALL DATA?
7996 034650' 002004                  BGE     70$                     ;YES
7997 034652' 062716 000002           ADD     #2,(SP)                 ;NEXT GOOD DATA PATTERN
7998 034656' 005722                  TST     (R2)+                   ;NEXT BUFFER ADDRESS
7999 034660' 000762                  BR      50$                     ;CONTINUE CHECKING
8000 034662' 004737 020166'  70$:    JSR     PC,REUNA                ;RESTORE OPERATIONAL MICROCODE
8001 034666'                   ENDSEG
8002 034666'                                                     10001$:
8003 034666' 104405                                                      TRAP    C$ESEG
8004 034670'                 ENDTST
8005 034670'                                                     L10136:
8006 034670' 104401                                                      TRAP    C$ETST
```

```
                                    .SBTTL  TEST 21: TRANSMIT DONE TEST

                                    ;*************************************************************************
                                    ;
                                    ;THE TRANSMIT STATE MACHINE INFORMS THE PORT MODULE PROCESSOR OF A
                                    ;'TRANSMIT DONE' CONDITION. IT DOES THIS BY GENERATING AN INTERRUPT WHENEVER
                                    ;IT FINISHES  TRANSMITTING A DATAGRAM. SINCE THE 'TRANSMIT DONE' INTERRUPT IS
                                    ;A NECESSARY CONDITION OF EVERY DATAGRAM TRANSMISSION, THIS TEST WILL USE THE
                                    ;INTERRUPT TO INDICATE THAT THE TRANSMIT STATE MACHINE IS FUNCTIONING.
                                    ;
                                    ;MICROCODE MODULE D MICROTEST #1 WILL BE USED FOR THIS TEST. IT SETS
                                    ;UP THE T-11 FOR AN INTERRUPT, STARTS A DATAGRAM LOOPBACK AND WAITS FOR A
                                    ;TRANSMIT INTERRUPT. THE T-11  WILL BE RELEASED FROM THE LOOP IF THE XMIT DONE
                                    ;INTERRUPT OCCURS. UPON RELEASE THE DNI BIT WILL BE SET IN PCSR0 SIGNALING
                                    ;THAT THE TEST IS COMPLETE.
                                    ;
                                    ;TEST SEQUENCE:
                                    ;       1-LOAD MICROCODE MODULE 'D' IF NOT ALREADY DONE SO
                                    ;       2-VERIFY THE MICROMONITOR IN THE 'IN MONITOR' STATE
                                    ;       3-SELECT MICROTEST #1
                                    ;       4-VERIFY 'DNI' BIT IN PCSR0 AFTER A RESONABLE PERIOD OF TIME
                                    ;       5-WRITE ONE TO CLEAR DNI
                                    ;
                                    ;*************************************************************************

 034672'                            BGNTST
 034672'                                                                            T21::
                                    ;
                                    ;CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                    ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'
                                    ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                    ;
 034672'                             BGNSEG
 034672' 104404                                                                     TRAP    C$BSEG
 034674' 022737  000104  000326'        CMP     #'D,MICRO           ;HAS MICROCODE MODULE 'D' BEEN LOADED
 034702' 001004                         BNE     5$                  ;NO
 034704' 122777  000001  143426         CMPB    #INMON,@PCSR1       ;YES, IS THE MICROMONITOR ACTIVE?
 034712' 001440                         BEQ     20$                 ;YES SKIP LOADING THE MICROMODULE
 034714' 012737  000104  000326' 5$:    MOV     #'D,MICRO           ;GO LOAD MICRO MODULE 'D'
 034722' 004737  020340'                JSR     PC,LODMIC
 034726' 103002                         BCC     10$                 ;OK
 034730'                                ESCAPE  TST
 034730' 104410                                                                     TRAP    C$ESCAPE
 034732' 000246                                                                     .WORD   L10137-.
 034734' 012737  000176  000332' 10$:   MOV     #2*SECOND,METER     ;WAIT FOR THE MICROMONITOR
 034742' 004737  017316'                JSR     PC,CHKDNI
 034746' 103022                         BCC     20$                 ;OK
 034750' 012737  001000' 000310'        MOV     #SDNI,BITNAM
 034756' 012737  001277' 000312'        MOV     #SNSET,BITSTA
 034764' 012737  001342' 000314'        MOV     #SAFTER,PWHEN
 034772' 012737  001357' 000316'        MOV     #SGTCMD,PCOMND
 035000'                                ERRHRD  100,TRNDON,MSG1
 035000' 104456                                                                     TRAP    C$ERHRD
 035002' 000144                                                                     .WORD   100
 035004' 002445'                                                                    .WORD   TRNDON
 035006' 012716'                                                                    .WORD   MSG1
```

```
035010'                                    ESCAPE  TST
035010' 104410                                                                                      TRAP    C$ESCAPE
035012' 000166                                                                                      .WORD   L10137-.
035014' 004737  017362'           20$:     JSR     PC,CLRDNI                   ;CLEAR DNI BIT
035020' 103006                             BCC     25$
035022'                                    ERRHRD  101,TRNDON,RACMG7           ;DNI DID NOT CLEAR!
035022' 104456                                                                                      TRAP    C$ERHRD
035024' 000145                                                                                      .WORD   101
035026' 002445'                                                                                     .WORD   TRNDON
035030' 012670'                                                                                     .WORD   RACMG7
035032'                                    ESCAPE TST
035032' 104410                                                                                      TRAP    C$ESCAPE
035034' 000144                                                                                      .WORD   L10137-.
035036'                           25$:
035036'                             ENDSEG
035036'                                                                                  10000$:
035036' 104405                                                                                      TRAP    C$ESEG
                                  ;
                                  ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE. LOAD THE COMMAND
                                  ;FIELD BITS OF PCSRO WITH A 1 TO START THE EXECUTION OF MICROTEST #1.
                                  ;WAIT ABOUT 1 SECOND FOR IT TO FINISH. IF NO 'DNI' SET PRINT ERROR.
                                  ;
035040'                             BGNSEG
035040' 104404                                                                                      TRAP    C$BSEG
035042' 004737  020060'                    JSR     PC,CHKMON                   ;WAIT FOR MICROMONITOR
035046' 103006                             BCC     30$                         ;OK
035050'                                    ERRHRD  102,TRNDON,MSG46            ;PRINT ERROR
035050' 104456                                                                                      TRAP    C$ERHRD
035052' 000146                                                                                      .WORD   102
035054' 002445'                                                                                     .WORD   TRNDON
035056' 016666'                                                                                     .WORD   MSG46
035060'                                    ESCAPE  TST                         ;LEAVE TEST
035060' 104410                                                                                      TRAP    C$ESCAPE
035062' 000116                                                                                      .WORD   L10137-.
035064' 012777  000001  143244    30$:     MOV     #1,@PCSRO                   ;TELL T11 TO EXECUTE FIRST MICROTEST
035072' 012737  000077  000332'            MOV     #1*SECOND,METER             ;WAIT FOR DNI
035100' 004737  017316'                    JSR     PC,CHKDNI
035104' 103025                             BCC     40$                         ;OK-'DNI' SET
                                                                               ;ERROR 'DNI' DID NOT SET! NO TRANSMIT
                                                                               ;INTERRUPT!
035106'                                    ERRHRD  103,TRNDON,MSG36            ;PRINT ERROR MESSAGE
035106' 104456                                                                                      TRAP    C$ERHRD
035110' 000147                                                                                      .WORD   103
035112' 002445'                                                                                     .WORD   TRNDON
035114' 015770'                                                                                     .WORD   MSG36
035116' 004737  020132'                    JSR     PC,CHKINT                   ;SEE IF ANY ERROR INTERRUPTS OCCURRED
035122' 103006                             BCC     35$                         ;NO, OK
035124'                                    ERRHRD  103,TRNDON,MSG44            ;PRINT ERROR MESSAGE
035124' 104456                                                                                      TRAP    C$ERHRD
035126' 000147                                                                                      .WORD   103
035130' 002445'                                                                                     .WORD   TRNDON
035132' 016442'                                                                                     .WORD   MSG44
035134'                                    ESCAPE  TST                         ;LEAVE TEST
035134' 104410                                                                                      TRAP    C$ESCAPE
035136' 000042                                                                                      .WORD   L10137-.
035140' 012702  000001            35$:     MOV     #1,R2                       ;MICROTEST #1 IS HUNG
```

```
035144'                                  ERRHRD  104,TRNDON,MSG12        ;PRINT MESSAGE ABOUT HUNG MICROTEST
035144' 104456                                                                   TRAP    C$ERHRD
035146' 000150                                                                   .WORD   104
035150' 002445'                                                                  .WORD   TRNDON
035152' 013466'                                                                  .WORD   MSG12
035154'                                  ESCAPE  TST
035154' 104410                                                                   TRAP    C$ESCAPE
035156' 000022                                                                   .WORD   L10137-.

                                 ;
                                 ;WRITE 1 TO CLEAR DNI BIT
                                 ;
035160' 004737  017362'          40$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
035164' 103004                           BCC     55$
035166'                                  ERRHRD  105,TRNDON,RACMG7       ;ERROR DNI DID NOT CLEAR!
035166' 104456                                                                   TRAP    C$ERHRD
035170' 000151                                                                   .WORD   105
035172' 002445'                                                                  .WORD   TRNDON
035174' 012670'                                                                  .WORD   RACMG7
035176'                          55$:
035176'                            ENDSEG
035176'                                                                  10001$:
035176' 104405                                                                   TRAP    C$ESEG
035200'                          ENDTST
035200'                                                                  L10137:
035200' 104401                                                                   TRAP    C$ETST
```

```
                                        .SBTTL  TEST 22: RECEIVER DONE TEST

                                        ;*********************************************************************
                                        ;
                                        ;THE LINK HARDWARE INCLUDES LOGIC TO TELL THE DEUNA PROCESSOR WHEN
                                        ;A LINK MEMORY BUFFER HAS BEEN FILLED AND DATA IS AVAILABLE FOR PROCESSING.
                                        ;THE HARDWARE INTERRUPTS THE DEUNA PROCESSOR. BECAUSE THE INTERRUPT HAPPENS
                                        ;WHEN A LINK MEMORY BUFFER IS FULL AND THE LINK MEMORY IS FILLED BY THE
                                        ;OPERATION OF THE RECEIVE STATE MACHINE, THE INTERRUPT CAN BE USED TO CHECK
                                        ;IF THE STATE MACHINE WORKS.
                                        ;
                                        ;MICROCODE MODULE D MICROTEST #2 WILL BE USED FOR THIS TEST. IT SETS
                                        ;UP THE T-11 FOR AN INTERRUPT, STARTS A DATAGRAM LOOPBACK AND WAITS FOR A
                                        ;RECEIVER INTERRUPT. THE T-11  WILL BE RELEASED FROM THE LOOP IF THE
                                        ;INTERRUPT OCCURS. UPON RELEASE THE DNI BIT WILL BE SET IN PCSRO SIGNALING
                                        ;THAT THE TEST IS COMPLETE.
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROCODE MODULE 'D' IF NOT ALREADY DONE SO
                                        ;       2-VERIFY THE MICROMONITOR IN THE 'IN MONITOR' STATE
                                        ;       3-SELECT MICROTEST #2
                                        ;       4-VERIFY 'DNI' BIT IN PCSRO AFTER A RESONABLE PERIOD OF TIME
                                        ;       5-WRITE ONE TO CLEAR DNI
                                        ;
                                        ;*********************************************************************

 035202'                                BGNTST
 035202'                                                                        T22::
                                        ;
                                        ;CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
 035202'                                  BGNSEG
 035202' 104404                                                                         TRAP    C$BSEG
 035204' 022737  000104  000326'                CMP     #'D,MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED
 035212' 001004                                 BNE     5$                      ;NO
 035214' 122777  000001  143116                 CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
 035222' 001440                                 BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
 035224' 012737  000104  000326' 5$:            MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
 035232' 004737  020340'                        JSR     PC,LODMIC
 035236' 103002                                 BCC     10$                     ;OK
 035240'                                        ESCAPE  TST
 035240' 104410                                                                         TRAP    C$ESCAPE
 035242' 000246                                                                         .WORD   L10140-.
 035244' 012737  000176  000332' 10$:           MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
 035252' 004737  017316'                        JSR     PC,CHKDNI
 035256' 103022                                 BCC     20$                     ;OK
 035260' 012737  001000' 000310'                MOV     #$DNI,BITNAM
 035266' 012737  001277' 000312'                MOV     #$NSET,BITSTA
 035274' 012737  001342' 000314'                MOV     #$AFTER,PWHEN
 035302' 012737  001357' 000316'                MOV     #$GTCMD,PCOMND
 035310'                                        ERRHRD  106,RCVDON,MSG1
 035310' 104456                                                                         TRAP    C$ERHRD
 035312' 000152                                                                         .WORD   106
 035314' 002477'                                                                        .WORD   RCVDON
```

```
035316' 012716'                                                                    .WORD   MSG1
035320'                                 ESCAPE  TST
035320' 104410                                                                     TRAP    C$ESCAPE
035322' 000166                                                                     .WORD   L10140-.
035324' 004737  017362'         20$:    JSR     PC,LLRDNI               ;CLEAR DNI BIT
035330' 103006                          BCC     25$
035332'                                 ERRHRD  107,RCVDON,RACMG7       ;DNI DID NOT CLEAR!
035332' 104456                                                                     TRAP    C$ERHRD
035334' 000153                                                                     .WORD   107
035336' 002477'                                                                    .WORD   RCVDON
035340' 012670'                                                                    .WORD   RACMG7
035342'                                 ESCAPE TST
035342' 104410                                                                     TRAP    C$ESCAPE
035344' 000144                                                                     .WORD   L10140-.
035346'                         25$:
035346'                           ENDSEG
035346'                                                                     10000$:
035346' 104405                                                                     TRAP    C$ESEG
                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE. LOAD THE COMMAND
                                ;FIELD BITS OF PCSRO WITH A 2 TO START THE EXECUTION OF MICROTEST #2.
                                ;WAIT ABOUT 1 SECOND FOR IT TO FINISH. IF NO 'DNI' SET PRINT ERROR.
                                ;
035350'                           BGNSEG
035350' 104404                                                                     TRAP    C$BSEG
035352' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
035356' 103006                          BCC     30$                     ;OK
035360'                                 ERRHRD  108,RCVDON,MSG46        ;PRINT ERROR
035360' 104456                                                                     TRAP    C$ERHRD
035362' 000154                                                                     .WORD   108
035364' 002477'                                                                    .WORD   RCVDON
035366' 016666'                                                                    .WORD   MSG46
035370'                                 ESCAPE  TST                     ;LEAVE TEST
035370' 104410                                                                     TRAP    C$ESCAPE
035372' 000116                                                                     .WORD   L10140-.
035374' 012777  000002  142734  30$:    MOV     #2,@PCSRO               ;TELL T11 TO EXECUTE MICROTEST #2
035402' 012737  000077  000332'         MOV     #1*SECOND,METER         ;WAIT FOR DNI
035410' 004737  017316'                 JSR     PC,CHKDNI
035414' 103025                          BCC     40$                     ;OK DNI SET
                                                                        ;ERROR-DNI NOT SET, NO RECEIVER
                                                                        ;INTERRUPT
035416'                                 ERRHRD  109,RCVDON,MSG37        ;PRINT ERROR MESSAGE
035416' 104456                                                                     TRAP    C$ERHRD
035420' 000155                                                                     .WORD   109
035422' 002477'                                                                    .WORD   RCVDON
035424' 016012'                                                                    .WORD   MSG37
035426' 004737  020132'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
035432' 103006                          BCC     35$                     ;NO, OK
035434'                                 ERRHRD  110,RCVDON,MSG44        ;PRINT ERROR MESSAGE
035434' 104456                                                                     TRAP    C$ERHRD
035436' 000156                                                                     .WORD   110
035440' 002477'                                                                    .WORD   RCVDON
035442' 016442'                                                                    .WORD   MSG44
035444'                                 ESCAPE  TST
035444' 104410                                                                     TRAP    C$ESCAPE
035446' 000042                                                                     .WORD   L10140-.
```

```
035450' 012702  000002          35$:    MOV     #2,R2                           ;MICROTEST #2 IS HUNG
035454'                                 ERRHRD  111,RCVDON,MSG12                ;PRINT ERROR MESSAGE
035454' 104456                                                                                  TRAP    C$ERHRD
035456' 000157                                                                                  .WORD   111
035460' 002477'                                                                                 .WORD   RCVDON
035462' 013466'                                                                                 .WORD   MSG12
035464'                                 ESCAPE  TST
035464' 104410                                                                                  TRAP    C$ESCAPE
035466' 000022                                                                                  .WORD   L10140-.
                                ;
                                ;WRITE 1 TO CLEAR DNI BIT
                                ;
035470' 004737  017362'         40$:    JSR     PC,CLRDNI                       ;CLEAR DNI BIT
035474' 103004                          BCC     55$
035476'                                 ERRHRD  112,RCVDON,RACMG7               ;ERROR DNI DID NOT CLEAR!
035476' 104456                                                                                  TRAP    C$ERHRD
035500' 000160                                                                                  .WORD   112
035502' 002477'                                                                                 .WORD   RCVDON
035504' 012670'                                                                                 .WORD   RACMG7
035506'                         55$:
035506'                           ENDSEG
035506'                                                                                 10001$:
035506' 104405                                                                                  TRAP    C$ESEG
035510'                         ENDTST
035510'                                                                                 L10140:
035510' 104401                                                                                  TRAP    C$ETST
```

```

                                                .SBTTL  TEST 23: DATA BYTE FRAMING TEST

                                                ;*********************************************************************
                                                ;
                                                ;THIS TEST WILL CHECK THE LINK MODULE DATA PATH FOR BYTE DATA BOUNDARY
                                                ;CONDITIONS.
                                                ;
                                                ;THE T-11 PROCESSOR WILL TRANSMIT DATA IN LOOPBACK MODE. THE DATA WILL BE
                                                ;ORGANIZED SUCH THAT DATA BOUNDARIES ARE CREATED BETWEEN ADJACENT BYTES
                                                ;IN THE DATA STREAM (I.E. 1111111110000000011...) THE T-11 PROCESSOR WILL
                                                ;VERIFY THE CONDITION OF THE DATA AFTER IT IS LOOPED BACK TO THE RECEIVER
                                                ;DATA BUFFER.
                                                ;
                                                ;THIS TEST WILL USE MICROCODE MODULE 'D' MICROTEST #3. TESTING OF THE DATA
                                                ;FRAMING WILL BE DONE BY THE T-11 PROCESSOR. THE HOST PROCESSOR, MEANWHILE,
                                                ;WILL WAIT FOR A 'DNI' IN REGISTER PCSR0. IF 'DNI' APPEARS, THE HOST PROCESSOR
                                                ;WILL CHECK PCSR1 FOR AN ERROR CONDITION. IF AN ERROR CONDITION IS SET,
                                                ;ADDITIONAL ERROR INFORMATION WILL BE FOUND IN THE PCBB AS FOLLOWS:
                                                ;
                                                ;PCBB+0: RECEIVER STATUS WORD
                                                ;PCBB+2: DATA TRANSMITTED
                                                ;PCBB+4: DATA RECEIVED
                                                ;PCBB+6: WORD OFFSET INTO RECEIVER BUFFER OF BAD DATA
                                                ;
                                                ;TEST SEQUENCE:
                                                ;       1-LOAD MICROCODE MODULE 'D' IF NOT ALREADY DONE SO
                                                ;       2-VERIFY MICROMONITOR TO ENTER THE 'INMON' STATE
                                                ;       3-CLEAR OUT THE PCBB+,+2,+4,+6
                                                ;       4-SELECT MICROTEST #3
                                                ;       5-VERIFY 'DNI' BIT SET IN PCSR0
                                                ;       6-CHECK FOR AN ERROR CONDITION IN PCSR1
                                                ;       7-WRITE ONE TO CLEAR THE DNI BIT
                                                ;
                                                ;*********************************************************************

 035512'                                        BGNTST
 035512'                                                                                        T23::

                                                ;
                                                ;CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                                ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'
                                                ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                                ;
 035512'                                          BGNSEG
 035512' 104404                                                                         TRAP    C$BSEG
 035514' 022737  000104  000326'        CMP     #'D,MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED
 035522' 001004                         BNE     5$                      ;NO
 035524' 122777  000001  142606         CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
 035532' 001440                         BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
 035534' 012737  000104  000326' 5$:    MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
 035542' 004737  020340'                JSR     PC,LODMIC
 035546' 103002                         BCC     10$                     ;OK
 035550'                                ESCAPE  TST
 035550' 104410                                                                         TRAP    C$ESCAPE
 035552' 000316                                                                         .WORD   L10141-.
 035554' 012737  000176  000332' 10$:   MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
```

```
035562' 004737  017316'                 JSR     PC,CHKDNI
035566' 103022                          BCC     20$                     ;OK
035570' 012737  001000' 000310'         MOV     #$DNI,BITNAM            ;ERROR 'DNI' BIT NEVER SET
035576' 012737  001277' 000312'         MOV     #$NSET,BITSTA
035604' 012737  001342' 000314'         MOV     #$AFTER,PWHEN
035612' 012737  001357' 000316'         MOV     #$GTCMD,PCOMND
035620'                                 ERRHRD  113,DBFRAM,MSG1
035620' 104456                                                                  TRAP    C$ERHRD
035622' 000161                                                                  .WORD   113
035624' 002531'                                                                 .WORD   DBFRAM
035626' 012716'                                                                 .WORD   MSG1
035630'                                 ESCAPE  TST
035630' 104410                                                                  TRAP    C$ESCAPE
035632' 000236                                                                  .WORD   L10141-.
035634' 004737  017362'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
035640' 103006                          BCC     25$
035642'                                 ERRHRD  114,DBFRAM,RACMG7       ;DNI DID NOT CLEAR!
035642' 104456                                                                  TRAP    C$ERHRD
035644' 000162                                                                  .WORD   114
035646' 002531'                                                                 .WORD   DBFRAM
035650' 012670'                                                                 .WORD   RACMG7
035652'                                 ESCAPE TST
035652' 104410                                                                  TRAP    C$ESCAPE
035654' 000214                                                                  .WORD   L10141-.
035656'                         25$:
035656'                           ENDSEG
035656'                                                                 10000$:
035656' 104405                                                                  TRAP    C$ESEG

                                ;
                                ;WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE. CLEAR THE PCBB. LOAD THE
                                ;COMMAND FIELD OF PCSRO WITH A 3 TO START THE EXECUTION OF MICROTEST #3.
                                ;WAIT FOR THE 'DNI' BIT. IF NO 'DNI' OR IF ANY EXTRANEOUS INTERRUPTS HAPPEN
                                ;PRINT ERROR
                                ;

035660'                           BGNSEG
035660' 104404                                                                  TRAP    C$BSEG
035662' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
035666' 103006                          BCC     30$                     ;OK
035670'                                 ERRHRD  115,DBFRAM,MSG46        ;PRINT ERROR
035670' 104456                                                                  TRAP    C$ERHRD
035672' 000163                                                                  .WORD   115
035674' 002531'                                                                 .WORD   DBFRAM
035676' 016666'                                                                 .WORD   MSG46
035700'                                 ESCAPE  TST                     ;LEAVE TEST
035700' 104410                                                                  TRAP    C$ESCAPE
035702' 000166                                                                  .WORD   L10141-.
035704' 005037  000606'         30$:    CLR     PCBB                    ;THIS IS WHERE MICROCODE WILL PUT...
                                                                        ;RECEIVE BUFFER STATUS WORD
035710' 005037  000610'                 CLR     PCBB+2                  ;HERE IS WHERE THE GOOD DATA GOES
035714' 005037  000612'                 CLR     PCBB+4                  ;HERE IS WHERE THE BAD DATA GOES
035720' 005037  000614'                 CLR     PCBB+6                  ;HERE IS WHERE THE BUFFER OFFSET GOES
035724' 012777  000003  142404          MOV     #3,@PCSRO               ;TELL T11 TO EXECUTE MICROTEST #3
035732' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
035740' 004737  017316'                 JSR     PC,CHKDNI
035744' 103021                          BCC     40$
```

```
035746' 004737  020132'                  JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
035752' 103006                           BCC     35$                     ;NO, OK
035754'                                  ERRHRD  116,DBFRAM,MSG44        ;PRINT ERROR MESSAGE
035754' 104456                                                                                   TRAP    C$ERHRD
035756' 000164                                                                                   .WORD   116
035760' 002531'                                                                                  .WORD   DBFRAM
035762' 016442'                                                                                  .WORD   MSG44
035764'                                  ESCAPE  TST
035764' 104410                                                                                   TRAP    C$ESCAPE
035766' 000102                                                                                   .WORD   L10141-.
035770' 012702  000003          35$:     MOV     #3,R2                   ;MICROTEST #3 IS HUNG
035774'                                  ERRHRD  117,DBFRAM,MSG12
035774' 104456                                                                                   TRAP    C$ERHRD
035776' 000165                                                                                   .WORD   117
036000' 002531'                                                                                  .WORD   DBFRAM
036002' 013466'                                                                                  .WORD   MSG12
036004'                                  ESCAPE  TST
036004' 104410                                                                                   TRAP    C$ESCAPE
036006' 000062                                                                                   .WORD   L10141-.
                                ;
                                ;OK WE GOT 'DNI' NOW CHECK PCSR1 TO SEE IF THE T-11 FOUND A BOGUS BYTE IN THE
                                ;RECEIVED DATA. IF SO, GET THE INFORMATION FROM THE PCBB AND PRINT ERROR
                                ;
036010' 122777  000003  142322  40$:     CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
036016' 001014                           BNE     50$                     ;NO
036020' 013701  000614'                  MOV     PCBB+6,R1               ;GET BUFFER OFFSET FOR ERROR REPORT
036024' 013702  000610'                  MOV     PCBB+2,R2               ;THIS IS THE GOOD DATA FOR ERROR REPORT
036030' 013703  000612'                  MOV     PCBB+4,R3               ;THIS IS THE BAD DATA FOR ERROR REPORT
036034' 013704  000606'                  MOV     PCBB,R4                 ;THIS IS THE RECEIVER STATUS WORD
036040'                                  ERRHRD  118,DBFRAM,MSG17        ;PRINT ERROR MESSAGE
036040' 104456                                                                                   TRAP    C$ERHRD
036042' 000166                                                                                   .WORD   118
036044' 002531'                                                                                  .WORD   DBFRAM
036046' 013730'                                                                                  .WORD   MSG17
                                ;
                                ;WRITE '1' TO CLEAR 'DNI' BIT
                                ;
036050' 004737  017362'         50$:     JSR     PC,CLRDNI               ;CLEAR DNI BIT
036054' 103004                           BCC     55$
036056'                                  ERRHRD  119,DBFRAM,RACMG7       ;ERROR DNI DID NOT CLEAR!
036056' 104456                                                                                   TRAP    C$ERHRD
036060' 000167                                                                                   .WORD   119
036062' 002531'                                                                                  .WORD   DBFRAM
036064' 012670'                                                                                  .WORD   RACMG7
036066'                         55$:
036066'                           ENDSEG
036066'                                                                          10001$:
036066' 104405                                                                                   TRAP    C$ESEG
036070'                         ENDTST
036070'                                                                          L10141:
036070' 104401                                                                                   TRAP    C$ETST
```

```
                                        .SBTTL  TEST 24: DATA WORD FRAMING TEST

                                        ;*********************************************************************
                                        ;
                                        ;THIS TEST WILL CHECK THE LINK MODULE DATA PATH FOR WORD DATA BOUNDARY
                                        ;CONDITIONS.
                                        ;
                                        ;THE T-11 PROCESSOR WILL TRANSMIT DATA IN LOOPBACK MODE. THE DATA WILL BE
                                        ;ORGANIZED SUCH THAT DATA BOUNDARIES ARE CREATED BETWEEN ADJACENT WORDS
                                        ;IN THE DATA STREAM (I.E. 1111111111111111000000000000000011...) THE T-11
                                        ;PROCESSOR WILL VERIFY THE CONDITION OF THE DATA AFTER IT IS LOOPED BACK TO
                                        ;THE RECEIVER DATA BUFFER.
                                        ;
                                        ;THIS TEST WILL USE MICROCODE MODULE 'D' MICROTEST #4. TESTING OF THE DATA
                                        ;FRAMING WILL BE DONE BY THE T-11 PROCESSOR. THE HOST PROCESSOR, MEANWHILE,
                                        ;WILL WAIT FOR A 'DNI' IN REGISTER PCSR0. IF 'DNI' APPEARS, THE HOST PROCESSOR
                                        ;WILL CHECK PCSR1 FOR AN ERROR CONDITION. IF AN ERROR CONDITION IS SET,
                                        ;ADDITIONAL ERROR INFORMATION WILL BE FOUND IN THE PCBB AS FOLLOWS:
                                        ;
                                        ;PCBB+0: RECEIVER STATUS WORD
                                        ;PCBB+2: DATA TRANSMITTED
                                        ;PCBB+4: DATA RECEIVED
                                        ;PCBB+6: WORD OFFSET INTO RECEIVER BUFFER OF BAD DATA
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROCODE MODULE 'D' IF NOT ALREADY DONE SO
                                        ;       2-VERIFY MICROMONITOR IS IN THE 'INMON' STATE
                                        ;       3-CLEAR OUT THE PCBB+0,+2,+4,+6
                                        ;       4-SELECT MICROTEST #4
                                        ;       5-VERIFY 'DNI' BIT SET IN PCSR0
                                        ;       6-CHECK FOR AN ERROR CONDITION IN PCSR1
                                        ;       7-WRITE ONE TO CLEAR THE DNI BIT
                                        ;
                                        ;*********************************************************************

036072'                                 BGNTST
036072'                                                                         T24::

                                        ;
                                        ;CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
036072'                                  BGNSEG
036072' 104404                                                                          TRAP    C$BSEG
036074' 022737  000104  000326'                 CMP     #'D,MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED
036102' 001004                                  BNE     5$                      ;NO
036104' 122777  000001  142226                  CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
036112' 001440                                  BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
036114' 012737  000104  000326' 5$:             MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
036122' 004737  020340'                         JSR     PC,LODMIC
036126' 103002                                  BCC     10$                     ;OK
036130'                                         ESCAPE  TST
036130' 104410                                                                          TRAP    C$ESCAPE
036132' 000316                                                                          .WORD   L10142-.
036134' 012737  000176  000332' 10$:            MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
036142' 004737  017316'                         JSR     PC,CHKDNI
```

```
036146' 103022                          BCC     20$                     ;OK
036150' 012737  001000' 000310'         MOV     #$DNI,BITNAM
036156' 012737  001277' 000312'         MOV     #$NSET,BITSTA
036164' 012737  001342' 000314'         MOV     #$AFTER,PWHEN
036172' 012737  001357' 000316'         MOV     #$G1CMD,PCOMND
036200'                                 ERRHRD  120,DWFRAM,MSG1
036200' 104456                                                                  TRAP    C$ERHRD
036202' 000170                                                                  .WORD   120
036204' 002567'                                                                 .WORD   DWFRAM
036206' 012716'                                                                 .WORD   MSG1
036210'                                 ESCAPE  TST
036210' 104410                                                                  TRAP    C$ESCAPE
036212' 000236                                                                  .WORD   L10142-.
036214' 004737  017362'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
036220' 103006                          BCC     25$
036222'                                 ERRHRD  121,DWFRAM,RACMG7       ;DNI DID NOT CLEAR!
036222' 104456                                                                  TRAP    C$ERHRD
036224' 000171                                                                  .WORD   121
036226' 002567'                                                                 .WORD   DWFRAM
036230' 012670'                                                                 .WORD   RACMG7
036232'                                 ESCAPE TST
036232' 104410                                                                  TRAP    C$ESCAPE
036234' 000214                                                                  .WORD   L10142-.
036236'                         25$:
036236'                           ENDSEG
036236'                                                                 10000$:
036236' 104405                                                                  TRAP    C$ESEG

                                ;
                                ;WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE. CLEAR THE PCBB. LOAD THE
                                ;COMMAND FIELD OF PCSRO WITH A 4 TO START THE EXECUTION OF MICROTEST #4.
                                ;WAIT FOR THE 'DNI' BIT. IF NO 'DNI' OR IF ANY EXTRANEOUS INTERRUPTS HAPPEN
                                ;PRINT ERROR
                                ;
036240'                           BGNSEG
036240' 104404                                                                  TRAP    C$BSEG
036242' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
036246' 103006                          BCC     30$                     ;OK
036250'                                 ERRHRD  122,DWFRAM,MSG46        ;PRINT ERROR
036250' 104456                                                                  TRAP    C$ERHRD
036252' 000172                                                                  .WORD   122
036254' 002567'                                                                 .WORD   DWFRAM
036256' 016666'                                                                 .WORD   MSG46
036260'                                 ESCAPE  TST                     ;LEAVE TEST
036260' 104410                                                                  TRAP    C$ESCAPE
036262' 000166                                                                  .WORD   L10142-.
036264' 005037  000606'         30$:    CLR     PCBB                    ;THIS IS WHERE MICROCODE WILL PUT...
                                                                        ;RECEIVE BUFFER STATUS WORD
036270' 005037  000610'                 CLR     PCBB+2                  ;HERE IS WHERE THE GOOD DATA GOES
036274' 005037  000612'                 CLR     PCBB+4                  ;HERE IS WHERE THE BAD DATA GOES
036300' 005037  000614'                 CLR     PCBB+6                  ;HERE IS WHERE THE BUFFER OFFSET GOES
036304' 012777  000004  142024          MOV     #4,@PCSRO               ;TELL T11 TO EXECUTE MICROTEST #4
036312' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
036320' 004737  017316'                 JSR     PC,CHKDNI
036324' 103021                          BCC     40$
036326' 004737  020132'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
036332' 103006                          BCC     35$                     ;NO, OK
```

```
036334'                                 ERRHRD  123,DWFRAM,MSG44        ;PRINT ERROR MESSAGE
036334' 104456                                                                  TRAP    C$ERHRD
036336' 000173                                                                  .WORD   123
036340' 002567'                                                                 .WORD   DWFRAM
036342' 016442'                                                                 .WORD   MSG44
036344'                                 ESCAPE  TST
036344' 104410                                                                  TRAP    C$ESCAPE
036346' 000102                                                                  .WORD   L10142-.
036350' 012702  000004          35$:    MOV     #4,R2                   ;MICROTEST #4 IS HUNG
036354'                                 ERRHRD  124,DWFRAM,MSG12
036354' 104456                                                                  TRAP    C$ERHRD
036356' 000174                                                                  .WORD   124
036360' 002567'                                                                 .WORD   DWFRAM
036362' 013466'                                                                 .WORD   MSG12
036364'                                 ESCAPE  TST
036364' 104410                                                                  TRAP    C$ESCAPE
036366' 000062                                                                  .WORD   L10142-.
                                ;
                                ;OK WE GOT 'DNI' NOW CHECK PCSR1 TO SEE IF THE T-11 DETECTED A BOGUS WORD
                                ;IN THE RECEIVER BUFFER. IF SO, PRINT ERROR.
                                ;
036370' 122777  000003  141742  40$:    CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
036376' 001014                          BNE     50$                     ;NO
036400' 013701  000614'                 MOV     PCBB+6,R1               ;GET BUFFER OFFSET FOR ERROR REPORT
036404' 013702  000610'                 MOV     PCBB+2,R2               ;THIS IS THE GOOD DATA FOR ERROR REPORT
036410' 013703  000612'                 MOV     PCBB+4,R3               ;THIS IS THE BAD DATA FOR ERROR REPORT
036414' 013704  000606'                 MOV     PCBB,R4                 ;THIS IS THE RECEIVER STATUS WORD
036420'                                 ERRHRD  125,DWFRAM,MSG17        ;PRINT ERROR MESSAGE
036420' 104456                                                                  TRAP    C$ERHRD
036422' 000175                                                                  .WORD   125
036424' 002567'                                                                 .WORD   DWFRAM
036426' 013730'                                                                 .WORD   MSG17
                                ;
                                ;WRITE '1' TO CLEAR 'DNI' BIT
                                ;
036430' 004737  017362'         50$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
036434' 103004                          BCC     55$
036436'                                 ERRHRD  126,DWFRAM,RACMG7       ;ERROR DNI DID NOT CLEAR!
036436' 104456                                                                  TRAP    C$ERHRD
036440' 000176                                                                  .WORD   126
036442' 002567'                                                                 .WORD   DWFRAM
036444' 012670'                                                                 .WORD   RACMG7
036446'                         55$:
036446'                           ENDSEG
036446'                                                                 10001$:
036446' 104405                                                                  TRAP    C$ESEG
036450'                         ENDTST
036450'                                                                 L10142:
036450' 104401                                                                  TRAP    C$ETST
```

```
                                       .SBTTL  TEST 25: DATA PATH PATTERN TEST

                                       ;*********************************************************************
                                       ;
                                       ;THIS TEST WILL CHECK THE LINK MODULE DATA PATH FOR ALL 'STUCK AT 0' AND
                                       ;'STUCK AT 1' ERRORS.
                                       ;
                                       ;THE T-11 PROCESSOR WILL TRANSMIT DATAGRAMS OF MAXIMUM LENGTH IN LOOPBACK MODE.
                                       ;THIS PATTERN LOOPBACK PROCEDURE WILL BE USED FOR ALL PATTERNS OF UP TO WORD
                                       ;WIDTH.
                                       ;
                                       ;THIS TEST USES MICROMODULE 'D' MICROTEST #5 TO DO THE TESTING. THE HOST
                                       ;PROCESSOR WILL PASS A DATA PATTERN TO THE T-11 PROCESSOR THROUGH THE PCBB.
                                       ;THE T-11 WILL FILL A XMIT BUFFER WITH THE DATA PATTERN AND TRANSMIT THE
                                       ;DATAGRAM OVER THE LOOPBACK. THE T-11 PROCESSOR WILL VERIFY THE PATTERN
                                       ;IN THE RECEIVER BUFFER. IF THE T-11 FINDS AN ERROR, IT WILL WRITE THE FAILING
                                       ;PATTERN TO THE PCBB ALONG WITH THE OFFSET INTO THE RECEIVER BUFFER AT WHICH
                                       ;THE PATTERN WAS FOUND. IT WILL INFORM THE HOST OF THE ERROR BY SETTING PCSR1
                                       ;TO AN ERROR CONDITION. THE PCBB IS FORMATTED AS FOLLOWS:
                                       ;
                                       ;PCBB+0: DATA PATTERN
                                       ;PCBB+2: RECEIVER STATUS WORD
                                       ;PCBB+4: BAD DATA PATTERN
                                       ;PCBB+6: OFFSET INTO RECEIVER BUFFER WHERE BAD DATA WAS FOUND
                                       ;
                                       ;*********************************************************************

 036452'                               BGNTST
 036452'                                                                                     T25::
                                       ;
                                       ;CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                       ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'
                                       ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                       ;
 036452'                                 BGNSEG
 036452' 104404                                                                              TRAP    C$BSEG
 036454' 022737  000104  000326'              CMP     #'D,MICRO              ;HAS MICROCODE MODULE 'D' BEEN LOADED
 036462' 001004                               BNE     5$                     ;NO
 036464' 122777  000001  141646               CMPB    #INMON,@PCSR1          ;YES, IS THE MICROMONITOR ACTIVE?
 036472' 001440                               BEQ     20$                    ;YES SKIP LOADING THE MICROMODULE
 036474' 012737  000104  000326' 5$:          MOV     #'D,MICRO              ;GO LOAD MICRO MODULE 'D'
 036502' 004737  020340'                      JSR     PC,LODMIC
 036506' 103002                               BCC     10$                    ;OK
 036510'                                      ESCAPE  TST
 036510' 104410                                                                              TRAP    C$ESCAPE
 036512' 000332                                                                              .WORD   L10143-.
 036514' 012737  000176  000332' 10$:         MOV     #2*SECOND,METER        ;WAIT FOR THE MICROMONITOR
 036522' 004737  017316'                      JSR     PC,CHKDNI
 036526' 103022                               BCC     20$                    ;OK
 036530' 012737  001000' 000310'              MOV     #SDNI,BITNAM
 036536' 012737  001277' 000312'              MOV     #SNSET,BITSTA
 036544' 012737  001342' 000314'              MOV     #SAFTER,PWHEN
 036552' 012737  001357' 000316'              MOV     #SGTCMD,PCOMND
 036560'                                      ERRHRD  127,DPPAT,MSG1
 036560' 104456                                                                              TRAP    C$ERHRD
 036562' 000177                                                                              .WORD   127
```

```
036564' 002625'                                                                  .WORD   DPPAT
036566' 012716'                                                                  .WORD   MSG1
036570'                                  ESCAPE  TST
036570' 104410                                                                   TRAP    C$ESCAPE
036572' 000252                                                                   .WORD   L10143-.
036574' 004737  017362'         20$:     JSR     PC,CLRDNI            ;CLEAR DNI BIT
036600' 103006                           BCC     25$
036602'                                  ERRHRD  128,DPPAT,RACMG7     ;DNI DID NOT CLEAR!
036602' 104456                                                                   TRAP    C$ERHRD
036604' 000200                                                                   .WORD   128
036606' 002625'                                                                  .WORD   DPPAT
036610' 012670'                                                                  .WORD   RACMG7
036612'                                  ESCAPE TST
036612' 104410                                                                   TRAP    C$ESCAPE
036614' 000230                                                                   .WORD   L10143-.
036616'                         25$:
036616'                           ENDSEG
036616'                                                                  10000$:
036616' 104405                                                                   TRAP    C$ESEG
                                ;
                                ;CLEAR OUT THE PCBB+2,+4,+6 AND LOAD PCBB+0 WITH A DATA PATTERN SELECTED
                                ;FROM A LIST OF STANDARD DATA PATTERNS
                                ;
036620' 005037  000610'                  CLR     PCBB+2               ;HERE IS WHERE MICROCODE WILL PUT...
                                                                      ;STATUS WORD
036624' 005037  000612'                  CLR     PCBB+4               ;HERE IS WHERE BAD DATA GOES
036630' 005037  000614'                  CLR     PCBB+6               ;HERE IS WHERE BUFFER OFFSET GOES
036634' 012701  000520'                  MOV     #PATERN,R1           ;GET ADDRESS OF DATA PATTERN TABLE
036640' 012705  000005                   MOV     #5,R5                ;# OF DATA PATTERNS
036644' 012137  000606'         27$:     MOV     (R1)+,PCBB           ;LOAD PCBB WITH A DATA PATTERN
036650'                           BGNSEG
036650' 104404                                                                   TRAP    C$BSEG
                                ;
                                ;WAIT FOR THE T-11 TO ENTER THE 'INMON' STATE AT WHICH TIME SELECT MICROTEST #5
                                ;BY LOADING PCSRO COMMAND FIELD BITS WITH A 5. WAIT FOR 'DNI'.
                                ;
036652' 004737  020060'                  JSR     PC,CHKMON            ;WAIT FOR MICROMONITOR
036656' 103006                           BCC     30$                  ;OK
036660'                                  ERRHRD  129,DPPAT,MSG46      ;PRINT ERROR
036660' 104456                                                                   TRAP    C$ERHRD
036662' 000201                                                                   .WORD   129
036664' 002625'                                                                  .WORD   DPPAT
036666' 016666'                                                                  .WORD   MSG46
036670'                                  ESCAPE  TST                  ;LEAVE TEST
036670' 104410                                                                   TRAP    C$ESCAPE
036672' 000152                                                                   .WORD   L10143-.
036674' 012777  000005  141434  30$:     MOV     #5,@PCSRO            ;TELL T11 TO EXECUTE MICROTEST #5
036702' 012737  000176  000332'          MOV     #2*SECOND,METER      ;WAIT FOR DNI
036710' 004737  017316'                  JSR     PC,CHKDNI
036714' 103021                           BCC     40$
036716' 004737  020132'                  JSR     PC,CHKINT            ;SEE IF ANY ERROR INTERRUPTS OCCURRED
036722' 103006                           BCC     35$                  ;NO, OK
036724'                                  ERRHRD  130,DPPAT,MSG44      ;PRINT ERROR MESSAGE
036724' 104456                                                                   TRAP    C$ERHRD
036726' 000202                                                                   .WORD   130
036730' 002625'                                                                  .WORD   DPPAT
```

```
036732' 016442'                                                                         .WORD   MSG44
036734'                                    ESCAPE  TST
036734' 104410                                                                          TRAP    C$ESCAPE
036736' 000106                                                                          .WORD   L10143-.
036740' 012702  000005          35$:       MOV     #5,R2                  ;MICROTEST #5 IS HUNG
036744'                                    ERRHRD  131,DPPAT,MSG12
036744' 104456                                                                          TRAP    C$ERHRD
036746' 000203                                                                          .WORD   131
036750' 002625'                                                                         .WORD   DPPAT
036752' 013466'                                                                         .WORD   MSG12
036754'                                    ESCAPE  TST
036754' 104410                                                                          TRAP    C$ESCAPE
036756' 000066                                                                          .WORD   L10143-.
                                ;
                                ;OK WE GOT 'DNI' NOW CHECK PCSR1 TO SEE IF AN ERROR HAPPENED, IF SO, PRINT
                                ;PERTINENT INFORMATION
                                ;
036760' 122777  000003  141352  40$:       CMPB    #INERR,@PCSR1          ;DID AN ERROR OCCUR?
036766' 001014                             BNE     50$                    ;NO
036770' 013701  000614'                    MOV     PCBB+6,R1              ;GET BUFFER OFFSET FOR ERROR REPORT
036774' 013702  000606'                    MOV     PCBB,R2                ;THIS IS THE GOOD DATA FOR ERROR REPORT
037000' 013703  000612'                    MOV     PCBB+4,R3              ;THIS IS THE BAD DATA FOR ERROR REPORT
037004' 013704  000610'                    MOV     PCBB+2,R4              ;THIS IS THE RECEIVER STATUS WORD
037010'                                    ERRHRD  132,DPPAT,MSG17 ;PRINT ERROR MESSAGE
037010' 104456                                                                          TRAP    C$ERHRD
037012' 000204                                                                          .WORD   132
037014' 002625'                                                                         .WORD   DPPAT
037016' 013730'                                                                         .WORD   MSG17
                                ;
                                ;WRITE '1' TO CLEAR 'DNI' BIT
                                ;
037020' 004737  017362'         50$:       JSR     PC,CLRDNI              ;CLEAR DNI BIT
037024' 103004                             BCC     55$
037026'                                    ERRHRD  133,DPPAT,RACMG7       ;ERROR DNI DID NOT CLEAR!
037026' 104456                                                                          TRAP    C$ERHRD
037030' 000205                                                                          .WORD   133
037032' 002625'                                                                         .WORD   DPPAT
037034' 012670'                                                                         .WORD   RACMG7
037036'                         55$:
037036'                           ENDSEG
037036'                                                                       10001$:
037036' 104405                                                                          TRAP    C$ESEG
037040' 005305                             DEC     R5                     ;HAVE WE TESTED WITH ALL DATA PATTERNS?
037042' 001300                             BNE     27$                    ;NOT YET
037044'                         ENDTST
037044'                                                                       L10143:
037044' 104401                                                                          TRAP    C$ETST
```

```
                                        .SBTTL  TEST 26: STATUS MUX VERIFICATION TEST

                                        ;*****************************************************************************
                                        ;
                                        ;THE LINK WRITES STATUS IN LINK MEMORY AFTER EACH TRANSMIT ATTEMPT. THE STATUS
                                        ;GIVES INFORMATION ABOUT THE ATTEMPTED OPERATION. THE STATUS INFORMATION IS
                                        ;WRITTEN INTO THE FIRST TWO LOCATIONS OF THE TRANSMIT BUFFER. THIS INFORMATION
                                        ;IS ACCESSIBLE TO THE T-11 BY SIMPLY READING IT FROM LINK MEMORY.
                                        ;
                                        ;THIS TEST WILL VERIFY THAT THE STATUS INFORMATION IS WRITTEN INTO THE FIRST
                                        ;LOCATION OF THE TRANSMIT BUFFER. THE TEST WILL ALSO CHECK THE SECOND WORD
                                        ;OF THE TRANSMIT BUFFER.
                                        ;
                                        ;THIS TEST WILL USE MICROMODULE 'D' MICROTEST #6.
                                        ;WHEN THE TEST IS STARTED, THE T-11 PROCESSOR WILL SET UP THE LINK FOR LOOPBACK
                                        ;OF A DATA PATTERN. A BACKGRAOUND PATTERN WILL BE WRITTEN INTO THE FIRST WORD
                                        ;OF THE TRANSMIT BUFFER. THIS WORD SHOULD BE OVER-WRITTEN BY THE STATUS WHEN
                                        ;THE BUFFER IS TRANSMITTED. THE SECOND WORD OF THE TRANSMIT BUFFER CAN NOT BE
                                        ;WRITTEN WITH A BACKGROUND BECAUSE IT MUST DESIGNATE THE TRANSMIT BYTE COUNT.
                                        ;
                                        ;WHEN THE DATAGRAM HAS BEEN LOOPED BACK, THE T-11 PROCESSOR WILL PASS THE FIRST
                                        ;TWO WORDS OF THE TRANSMIT BUFFER TO THE HOST THRU THE PCBB+0 AND PCBB+2.
                                        ;
                                        ;PCBB+0: FIRST WORD OF TRANSMIT BUFFER
                                        ;PCBB+2: SECOND WORD OF TRANSMIT BUFFER
                                        ;
                                        ;THE CORRECT STATUS SHOULD BE:
                                        ;
                                        ;       TRANSMIT STATUS WORD 0 BITS 15,09:00 SHOULD BE ALL 0 AND
                                        ;                              BIT 13 SHOULD BE A 1
                                        ;
                                        ;       TRANSMIT STATUS WORD 1 BITS 15:13 SHOULD ALL BE 0
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO
                                        ;       2-VERIFY MICROMONITOR IS IN THE 'INMON' STATE
                                        ;       3-CLEAR PCBB+0 AND PCBB+2
                                        ;       4-SELECT MICROTEST #6
                                        ;       5-VERIFY 'DNI' SET
                                        ;       6-VERIFY PCBB+0 BITS 15,09:00 = 0 AND BIT 13 = 1
                                        ;         AND PCBB+2 BITS 15:13 = 0
                                        ;       7-WRITE ONE TO CLEAR 'DNI'
                                        ;
                                        ;*****************************************************************************

     037046'                            BGNTST
     037046'                                                                  T26::
                                        ;
                                        ;CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
     037046'                              BGNSEG
     037046' 104404                                                                        TRAP    C$BSEG
     037050' 022737  000104  000326'            CMP     #'D.MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED
     037056' 001004                             BNE     5$                      ;NO
```

```
037060' 122777  000001  141252          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
037066' 001440                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
037070' 012737  000104  000326' 5$:     MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
037076' 004737  020340'                 JSR     PC,LODMIC
037102' 103002                          BCC     10$                     ;OK
037104'                                 ESCAPE  TST
037104' 104410                                                                                  TRAP    C$ESCAPE
037106' 000500                                                                                  .WORD   L10144-.
037110' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
037116' 004737  017316'                 JSR     PC,CHKDNI
037122' 103022                          BCC     20$                     ;OK
037124' 012737  001000' 000310'         MOV     #$DNI,BITNAM
037132' 012737  001277' 000312'         MOV     #$NSET,BITSTA
037140' 012737  001342' 000314'         MOV     #$AFTER,PWHEN
037146' 012737  001357' 000316'         MOV     #$GTCMD,PCOMND
037154'                                 ERRHRD  134,STAMUX,MSG1
037154' 104456                                                                                  TRAP    C$ERHRD
037156' 000206                                                                                  .WORD   134
037160' 002663'                                                                                 .WORD   STAMUX
037162' 012716'                                                                                 .WORD   MSG1
037164'                                 ESCAPE  TST
037164' 104410                                                                                  TRAP    C$ESCAPE
037166' 000420                                                                                  .WORD   L10144-.
037170' 004737  017362'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
037174' 103006                          BCC     25$
037176'                                 ERRHRD  135,STAMUX,RACMG7       ;DNI DID NOT CLEAR!
037176' 104456                                                                                  TRAP    C$ERHRD
037200' 000207                                                                                  .WORD   135
037202' 002663'                                                                                 .WORD   STAMUX
037204' 012670'                                                                                 .WORD   RACMG7
037206'                                 ESCAPE TST
037206' 104410                                                                                  TRAP    C$ESCAPE
037210' 000376                                                                                  .WORD   L10144-.
037212'                         25$:
037212'                           ENDSEG
037212'                                                                                 10000$:
037212' 104405                                                                                  TRAP    C$ESEG

                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE, THEN CLEAR OUT THE
                                ;FIRST TWO WORDS OF THE PCBB. EXECUTE MICROTEST #6 BY LOADING THE COMMAND
                                ;FIELD BITS OF PCSR0 WITH A 6, WAIT FOR 'DNI'.
                                ;
037214'                           BGNSEG
037214' 104404                                                                                  TRAP    C$BSEG
037216' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
037222' 103006                          BCC     30$                     ;OK
037224'                                 ERRHRD  136,STAMUX,MSG46        ;PRINT ERROR
037224' 104456                                                                                  TRAP    C$ERHRD
037226' 000210                                                                                  .WORD   136
037230' 002663'                                                                                 .WORD   STAMUX
037232' 016666'                                                                                 .WORD   MSG46
037234'                                 ESCAPE  TST                     ;LEAVE TEST
037234' 104410                                                                                  TRAP    C$ESCAPE
037236' 000350                                                                                  .WORD   L10144-.
037240' 005037  000606'         30$:    CLR     PCBB                    ;THIS IS WHERE THE MICROCODE WILL...
                                                                        ;PUT THE CONTENTS OF THE FIRST WORD
```

```
                                                                   ;OF THE TRANSMIT BUFFER
037244' 005037  000610'                   CLR     PCBB+2           ;HERE IS WHERE THE SECOND WORD GOES
037250' 012777  000006  141060            MOV     #6,@PCSR0        ;TELL T11 TO EXECUTE MICROTEST #6
037256' 012737  000176  000332'           MOV     #2*SECOND,METER  ;WAIT FOR DNI
037264' 004737  017316'                   JSR     PC,CHKDNI
037270' 103021                            BCC     45$
037272' 004737  020132'                   JSR     PC,CHKINT        ;SEE IF ANY ERROR INTERRUPTS OCCURRED
037276' 103006                            BCC     35$              ;NO, OK
037300'                                   ERRHRD  137,STAMUX,MSG44 ;PRINT ERROR MESSAGE
037300' 104456                                                                          TRAP    C$ERHRD
037302' 000211                                                                          .WORD   137
037304' 002663'                                                                         .WORD   STAMUX
037306' 016442'                                                                         .WORD   MSG44
037310'                                   ESCAPE  TST
037310' 104410                                                                          TRAP    C$ESCAPE
037312' 000274                                                                          .WORD   L10144-.
037314' 012702  000006          35$:      MOV     #6,R2            ;MICROTEST #6 IS HUNG
037320'                                   ERRHRD  138,STAMUX,MSG12
037320' 104456                                                                          TRAP    C$ERHRD
037322' 000212                                                                          .WORD   138
037324' 002663'                                                                         .WORD   STAMUX
037326' 013466'                                                                         .WORD   MSG12
037330'                                   ESCAPE  TST
037330' 104410                                                                          TRAP    C$ESCAPE
037332' 000254                                                                          .WORD   L10144-.
                                ;
                                ;OK, NOW PCBB+0 SHOULD CONTAIN THE TRANSMIT STATUS WORD 0 AND PCBB+2 SHOULD
                                ;CONTAIN TRANSMIT STATUS WORD 1. CHECK THAT PCBB+0 CONTAINS STATUS BITS
                                ;FOR A GOOD TRANSMIT I.E. BITS 09:00 SHOULD ALL BE 0 BIT 13 SHOULD BE A 1
                                ;AND BIT 15 SHOULD BE A 0
                                ;
037334' 005001                  45$:      CLR     R1               ;WE ARE GOING TO CHECK TRANSMIT WORD 0
037336' 012703  000460'                   MOV     #BNAMT2,R3       ;POINT TO A TABLE OF BIT MNEMONICS
037342' 012704  000012                    MOV     #10.,R4          ;FIRST 10 BITS SHOULD BE 0
037346' 012702  000001                    MOV     #1,R2            ;R2 POINTS TO THE BIT WE ARE TESTING
037352' 030237  000606'         50$:      BIT     R2,PCBB          ;IS THIS BIT A 0?
037356' 001406                            BEQ     55$              ;YES
037360' 011337  000310'                   MOV     (R3),BITNAM      ;NO, GET POINTER TO BIT NAME ASCII STRING
037364'                                   ERRHRD  139,STAMUX,MSG18 ;PRINT ERROR MESSAGE
037364' 104456                                                                          TRAP    C$ERHRD
037366' 000213                                                                          .WORD   139
037370' 002663'                                                                         .WORD   STAMUX
037372' 014022'                                                                         .WORD   MSG18
037374' 062703  000002          55$:      ADD     #2,R3            ;POINT TO NEXT BIT MNEMONIC
037400' 006302                            ASL     R2               ;POINT TO NEXT BIT
037402' 005304                            DEC     R4               ;HAVE WE DONE ALL 10 BITS?
037404' 001362                            BNE     50$              ;NO

037406' 032737  020000  000606'           BIT     #BIT13,PCBB      ;IS BIT 13 A 1?
037414' 001007                            BNE     60$              ;YES
037416' 012737  001147' 000310'           MOV     #$BIT13,BITNAM   ;NO, POINT TO ASCII STRING
037424'                                   ERRHRD  140,STAMUX,MSG18 ;PRINT ERROR MESSAGE
037424' 104456                                                                          TRAP    C$ERHRD
037426' 000214                                                                          .WORD   140
037430' 002663'                                                                         .WORD   STAMUX
037432' 014022'                                                                         .WORD   MSG18
```

```

037434' 032737  100000  000606' 60$:    BIT     #BIT15,PCBB             ;IS BIT 15 A 0?
037442' 001407                          BEQ     65$                     ;YES
037444' 012737  001131' 000310'         MOV     #$BIT15,BITNAM          ;NO, POINT TO ASCII STRING
037452'                                 ERRHRD  141,STAMUX,MSG18        ;PRINT ERROR MESSAGE
037452' 104456                                                                  TRAP    C$ERHRD
037454' 000215                                                                  .WORD   141
037456' 002663'                                                                 .WORD   STAMUX
037460' 014022'                                                                 .WORD   MSG18
                                ;
                                ;NOW CHECK TRANSMIT WORD 1 BITS 15,14 AND 13 TO BE ALL 0
                                ;
037462' 005201                  65$:    INC     R1                      ;WE ARE CHECKING TRNASMIT WORD 1 NOW
037464' 032737  020000  000610'         BIT     #BIT13,PCBB+2           ;IS BIT 13 A 0?
037472' 001407                          BEQ     70$                     ;YES
037474' 012737  001147' 000310'         MOV     #$BIT13,BITNAM          ;NO, POINT TO ASCII STRING
037502'                                 ERRHRD  142,STAMUX,MSG18        ;PRINT ERROR MESSAGE
037502' 104456                                                                  TRAP    C$ERHRD
037504' 000216                                                                  .WORD   142
037506' 002663'                                                                 .WORD   STAMUX
037510' 014022'                                                                 .WORD   MSG18

037512' 032737  040000  000610' 70$:    BIT     #BIT14,PCBB+2           ;IS BIT 14 A 0?
037520' 001407                          BEQ     75$                     ;YES
037522' 012737  001140' 000310'         MOV     #$BIT14,BITNAM          ;NO, POINT TO ASCII STRING
037530'                                 ERRHRD  143,STAMUX,MSG18        ;PRINT ERROR MESSAGE
037530' 104456                                                                  TRAP    C$ERHRD
037532' 000217                                                                  .WORD   143
037534' 002663'                                                                 .WORD   STAMUX
037536' 014022'                                                                 .WORD   MSG18

037540' 032737  100000  000610' 75$:    BIT     #BIT15,PCBB+2           ;IS BIT 15 A 0?
037546' 001407                          BEQ     80$                     ;YES
037550' 012737  001131' 000310'         MOV     #$BIT15,BITNAM          ;NO, POINT TO ASCII STRING
037556'                                 ERRHRD  144,STAMUX,MSG18        ;PRINT ERROR MESSAGE
037556' 104456                                                                  TRAP    C$ERHRD
037560' 000220                                                                  .WORD   144
037562' 002663'                                                                 .WORD   STAMUX
037564' 014022'                                                                 .WORD   MSG18
                                ;
                                ;WRITE '1' TO CLEAR 'DNI' BIT
                                ;
037566' 004737  017362'         80$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
037572' 103004                          BCC     85$
037574'                                 ERRHRD  145,STAMUX,RACMG7       ;ERROR DNI DID NOT CLEAR!
037574' 104456                                                                  TRAP    C$ERHRD
037576' 000221                                                                  .WORD   145
037600' 002663'                                                                 .WORD   STAMUX
037602' 012670'                                                                 .WORD   RACMG7
037604'                         85$:
037604'                           ENDSEG
037604'                                                                 10001$:
037604' 104405                                                                  TRAP    C$ESEG
037606'                         ENDTST
037606'                                                                 L10144:
037606' 104401                                                                  TRAP    C$ETST
```

```
                                .SBTTL  TEST 27: LINK BYTE COUNTER TEST

                                ;*****************************************************************************
                                ;
                                ;BYTE COUNTERS ARE INVOLVED BOTH WITH THE LINK TRANSMIT FUNCTION AND THE LINK
                                ;RECEIVE FUNCTION. WHEN THE T-11 PREPARES A TRANSMIT BUFFER FOR TRANSMISSION
                                ;OF A DATAGRAM, IT WRITES THE INTENDED BYTE COUNT IN THE SECOND WORD OF THE
                                ;TRANSMIT BUFFER. WHEN TRANSMISSION OF THE TRANSMIT BUFFER BEGINS, THE BYTE
                                ;COUNT VALUE IS USED TO LOAD THE TRANSMIT BYTE COUNTER. THIS COUNTER IS
                                ;DECREMENTED BY THE TRANSMIT STATE MACHINE AS THE DATAGRAM IS TRANSMITTED.
                                ;THE DATAGRAM TRANSMISSION WILL CONTINUE UNTIL THE BYTE COUNTER IS DECREMENTED
                                ;TO ZERO.
                                ;
                                ;THE RECEIVER ALSO HAS A BYTE COUNTER. THIS COUNTER IS CLEARED AT THE START OF
                                ;A DATAGRAM RECEPTION AND IS INCREMENTED BY THE RECEIVE STATE MACHINE AS THE
                                ;DATAGRAM IS RECEIVED. THE VALUE IN THIS COUNTER IS WRITTEN INTO WORD TWO
                                ;OF THE RECEIVE BUFFER AT THE END OF RECEPTION.
                                ;
                                ;THIS TEST WILL USE MICROMODULE 'D' MICROTEST #7.
                                ;THIS TEST WILL VERIFY THE BYTE COUNT LOGIC BY LOOPING MESSAGES AND VERIFYING
                                ;THAT THE BYTE COUNT APPEARING IN THE RECEIVE BUFFER CORRESPONDS TO THE BYTE
                                ;COUNT THAT WAS WRITTEN TO THE TRANSMIT BYTE COUNTER. THE TEST WILL ALSO
                                ;VERIFY THAT THE ACTUAL NUMBER OF BYTES TRANSFERRED TO THE RECEIVE BUFFER
                                ;CORRESPONDS TO THE INTENDED BYTE COUNT.
                                ;
                                ;THE TRANSMIT BYTE COUNT IS PASSED TO THE T-11 VIA THE PCBB+0. AFTER THE
                                ;DATAGRAM LOOPBACK THE RECEIVE BYTE COUNT IS PLACED INTO PCBB+2 BY THE T-11
                                ;PROCESSOR. PCBB+4 IS LOADED BY THE T-11 PROCESSOR WITH THE ACTUAL NUMBER OF
                                ;BYTES THAT WERE TRANSFERRED TO THE RECEIVER BUFFER.
                                ;
                                ;PCBB+0: TRANSMIT BYTE COUNT
                                ;PCBB+2: RECEIVE BYTE COUNT
                                ;PCBB+4: ACTUAL NUMBER OF BYTES RECEIVED
                                ;
                                ;TEST SEQUENCE:
                                ;       1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO
                                ;       2-WRITE MINIMUM BYTE COUNT IN PCBB+0
                                ;       3-CLEAR PCBB+2 AND PCBB+4
                                ;       4-VERIFY MICROMONITOR IS IN 'INMON' STATE
                                ;       5-SELECT MICROTEST #7
                                ;       6-VERIFY TRANSMIT BYTE COUNT SAME AS RECEIVE BYTE COUNT
                                ;       7-VERIFY ACTUAL NUMBER OF BYTES RECEIVED IS CORRECT
                                ;       8-WRITE '1' TO CLEAR 'DNI'
                                ;       9-ADD 2 TO BYTE COUNT IN PCBB+0
                                ;       10-REPEAT STEPS 3-9 UNTIL PCBB+0 REACHES MAXIMUM BYTE COUNT
                                ;
                                ;*****************************************************************************

  037610'                       BGNTST
  037610'                                                                       T27::

                                ;
                                ;CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'
                                ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                ;
  037610'                        BGNSEG
```

```
037610' 104404                                                                      TRAP    C$BSEG
037612' 022737  000104  000326'          CMP     #'D,MICRO          ;HAS MICROCODE MODULE 'D' BEEN LOADED
037620' 001004                           BNE     5$                 ;NO
037622' 122777  000001  140510           CMPB    #!NMON,@PCSR1      ;YES, IS THE MICROMONITOR ACTIVE?
037630' 001440                           BEQ     20$                ;YES SKIP LOADING THE MICROMODULE
037632' 012737  000104  000326' 5$:      MOV     #'D,MICRO          ;GO LOAD MICRO MODULE 'D'
037640' 004737  020340'                  JSR     PC,LODMIC
037644' 103002                           BCC     10$                ;OK
037646'                                  ESCAPE  TST
037646' 104410                                                                      TRAP    C$ESCAPE
037650' 000334                                                                      .WORD   L10145-.
037652' 012737  000176  000332' 10$:     MOV     #2*SECOND,METER    ;WAIT FOR THE MICROMONITOR
037660' 004737  017316'                  JSR     PC,CHKDNI
037664' 103022                           BCC     20$                ;OK
037666' 012737  001000' 000310'          MOV     #$DNI,BITNAM
037674' 012737  001277' 000312'          MOV     #$NSET,BITSTA
037702' 012737  001342' 000314'          MOV     #$AFTER,PWHEN
037710' 012737  001357' 000316'          MOV     #$GTCMD,PCOMND
037716'                                  ERRHRD  146,LNKBYT,MSG1
037716' 104456                                                                      TRAP    C$ERHRD
037720' 000222                                                                      .WORD   146
037722' 002727'                                                                     .WORD   LNKBYT
037724' 012716'                                                                     .WORD   MSG1
037726'                                  ESCAPE  TST
037726' 104410                                                                      TRAP    C$ESCAPE
037730' 000254                                                                      .WORD   L10145-.
037732' 004737  017362'         20$:     JSR     PC,CLRDNI          ;CLEAR DNI BIT
037736' 103006                           BCC     25$
037740'                                  ERRHRD  147,LNKBYT,RACMG7  ;DNI DID NOT CLEAR!
037740' 104456                                                                      TRAP    C$ERHRD
037742' 000223                                                                      .WORD   147
037744' 002727'                                                                     .WORD   LNKBYT
037746' 012670'                                                                     .WORD   RACMG7
037750'                                  ESCAPE TST
037750' 104410                                                                      TRAP    C$ESCAPE
037752' 000232                                                                      .WORD   L10145-.
037754'                         25$:
037754'                           ENDSEG
037754'                                                                     10000$:
037754' 104405                                                                      TRAP    C$ESEG

                                ;
                                ;LOAD PCBB+0 WITH THE MINIMUM BYTE COUNT
                                ;
037756' 012737  000100  000606'          MOV     #MINBYT,PCBB       ;BEGIN WITH MINIMUM BYTE COUNT
                                ;
                                ;CLEAR PCBB+2 AND PCBB+4, AIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                ;EXECUTE MICROTEST #7 BY LOADING THE COMMAND FIELD OF PCSRO WITH A 7.
                                ;WAIT FOR 'DNI'
                                ;
037764'                         30$:
037764'                           BGNSEG
037764' 104404                                                                      TRAP    C$BSEG
037766' 005037  000610'                  CLR     PCBB+2             ;THIS IS WHERE MICROCODE WILL PUT...
                                                                    ;RECEIVER BYTE COUNT
037772' 005037  000612'                  CLR     PCBB+4             ;HERE IS WHERE MICROCODE WILL PUT...
                                                                    ;ACTUAL NUMBER OF BYTES RECEIVED
```

```
037776' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
040002' 103006                          BCC     35$                     ;OK
040004'                                 ERRHRD  148,LNKBYT,MSG46        ;PRINT ERROR
040004' 104456                                                                  TRAP    C$ERHRD
040006' 000224                                                                  .WORD   148
040010' 002727'                                                                 .WORD   LNKBYT
040012' 016666'                                                                 .WORD   MSG46
040014'                                 ESCAPE  TST                     ;LEAVE TEST
040014' 104410                                                                  TRAP    C$ESCAPE
040016' 000166                                                                  .WORD   L10145-.
040020' 012777  000007  140310  35$:    MOV     #7,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #7
040026' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
040034' 004737  017316'                 JSR     PC,CHKDNI
040040' 103021                          BCC     40$
040042' 004737  020132'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
040046' 103006                          BCC     36$                     ;NO, OK
040050'                                 ERRHRD  149,LNKBYT,MSG44        ;PRINT ERROR MESSAGE
040050' 104456                                                                  TRAP    C$ERHRD
040052' 000225                                                                  .WORD   149
040054' 002727'                                                                 .WORD   LNKBYT
040056' 016442'                                                                 .WORD   MSG44
040060'                                 ESCAPE  TST
040060' 104410                                                                  TRAP    C$ESCAPE
040062' 000122                                                                  .WORD   L10145-.
040064' 012702  000007          36$:    MOV     #7,R2                   ;MICROTEST #7 IS HUNG
040070'                                 ERRHRD  150,LNKBYT,MSG12
040070' 104456                                                                  TRAP    C$ERHRD
040072' 000226                                                                  .WORD   150
040074' 002727'                                                                 .WORD   LNKBYT
040076' 013466'                                                                 .WORD   MSG12
040100'                                 ESCAPE  TST
040100' 104410                                                                  TRAP    C$ESCAPE
040102' 000102                                                                  .WORD   L10145-.
                                ;
                                ;OK, CHECK PCBB+2, WHICH CONTAINS THE BYTE COUNT WRITTEN INTO THE RECEIVER
                                ;BUFFER BY THE LINK, TO SEE IF IT IS THE SAME AS THE TRANSMIT BYTE COUNT. ALSO
                                ;CHECK PCBB+4, WHICH IS THE ACTUAL NUMBER OF BYTES RECEIVED IN THE RECEIVER
                                ;BUFFER, TO SEE IF IT IS THE SAME AS THE RECEIVE BYTE COUNT
                                ;IF NOT PRINT AN ERROR
                                ;
040104' 013701  000606'         40$:    MOV     PCBB,R1                 ;GET TRANSMIT BYTE COUNT
040110' 010102                          MOV     R1,R2
040112' 013703  000610'                 MOV     PCBB+2,R3               ;GET RECEIVE BYTE COUNT
040116' 013704  000612'                 MOV     PCBB+4,R4               ;GET ACTUAL NUMBER OF BYTES TRANSMITTED
040122' 020203                          CMP     R2,R3                   ;IS THE RECEIVE BYTE COUNT CORRECT?
040124' 001002                          BNE     45$                     ;NO, ERROR
040126' 020104                          CMP     R1,R4                   ;IS THE ACTUAL NUMBER OF BYTES CORRECT?
040130' 001404                          BEQ     50$                     ;YES
040132'                         45$:    ERRHRD  151,LNKBYT,MSG19        ;PRINT ERROR MESSAGE
040132' 104456                                                                  TRAP    C$ERHRD
040134' 000227                                                                  .WORD   151
040136' 002727'                                                                 .WORD   LNKBYT
040140' 014052'                                                                 .WORD   MSG19
040142'                         50$:
040142'                                 ENDSEG
040142'                                                                 10001$:
```

```
040142' 104405                                                                               TRAP     C$ESEG
                                   ;
                                   ;WRITE '1' TO CLEAR THE 'DNI' BIT
                                   ;
040144' 004737  017362'                  JSR     PC,CLRDNI                 ;CLEAR DNI BIT
040150' 103006                           BCC     60$
040152'                                  ERRHRD  152,LNKBYT,RACMG7         ;ERROR DNI DID NOT CLEAR!
040152' 104456                                                                               TRAP     C$ERHRD
040154' 000230                                                                               .WORD    152
040156' 002727'                                                                              .WORD    LNKBYT
040160' 012670'                                                                              .WORD    RACMG7
040162'                                  ESCAPE  TST
040162' 104410                                                                               TRAP     C$ESCAPE
040164' 000020                                                                               .WORD    L10145-.
                                   ;
                                   ;CONTINUE CALLING MICROTEST #7 EACH TIME PASSING A LARGER (EVEN) BYTE COUNT
                                   ;UNTIL THE BYTE COUNT REACHES THE MAXIMUM COUNT ALLOWED
                                   ;
040166' 062737  000002  000606' 60$:     ADD     #2,PCBB                   ;UP THE BYTE COUNT
040174' 023727  000606' 002756           CMP     PCBB,#MAXBYT              ;HAVE WE DONE ALL EVEN BYTE COUNTS?
040202' 003670                           BLE     30$                       ;NO, CONTINE TESTING WITH NEW BYTE COUNT
040204'                            ENDTST
040204'                                                                         L10145:
040204' 104401                                                                               TRAP     C$ETST
```

```
                                        .SBTTL  TEST 28: ODD BYTE TEST

                                        ;*************************************************************************
                                        ;
                                        ;THIS TEST WILL VERIFY THAT THE LINK CAN TRANSMIT AND RECEIVE DATAGRAMS
                                        ;HAVING ONLY ODD BYTE COUNTS.
                                        ;
                                        ;THIS TEST IS IDENTICAL TO THE PREVIOUS BYTE COUNTER TEST WITH THE ONLY
                                        ;EXCEPTION THAT IT PASSES ONLY ODD BYTE COUNTS TO THE MICROCODE. IT ALSO
                                        ;USES MICROMODULE 'D' MICROTEST #7
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO
                                        ;       2-WRITE MINIMUM ODD BYTE COUNT IN PCBB+0
                                        ;       3-CLEAR PCBB+2 AND PCBB+4
                                        ;       4-VERIFY MICROMONITOR IS IN 'INMON' STATE
                                        ;       5-SELECT MICROTEST #7
                                        ;       6-VERIFY TRANSMIT BYTE COUNT SAME AS RECEIVE BYTE COUNT
                                        ;       7-VERIFY ACTUAL NUMBER OF BYTES RECEIVED IS CORRECT
                                        ;       8-WRITE '1' TO CLEAR 'DNI'
                                        ;       9-ADD 2 TO BYTE COUNT IN PCBB+0
                                        ;       10-REPEAT STEPS 3-9 UNTIL PCBB+0 REACHES MAXIMUM BYTE COUNT
                                        ;
                                        ;*************************************************************************

 040206'                                BGNTST
 040206'                                                                        T28::
                                        ;
                                        ;CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
 040206'                                 BGNSEG
 040206' 104404                                                                                 TRAP    C$BSEG
 040210' 022737  000104  000326'              CMP     #'D,MICRO              ;HAS MICROCODE MODULE 'D' BEEN LOADED
 040216' 001004                               BNE     5$                     ;NO
 040220' 122777  000001  140112               CMPB    #INMON,@PCSR1          ;YES, IS THE MICROMONITOR ACTIVE?
 040226' 001440                               BEQ     20$                    ;YES SKIP LOADING THE MICROMODULE
 040230' 012737  000104  000326' 5$:          MOV     #'D,MICRO              ;GO LOAD MICRO MODULE 'D'
 040236' 004737  020340'                      JSR     PC,LODMIC
 040242' 103002                               BCC     10$                    ;OK
 040244'                                      ESCAPE  TST
 040244' 104410                                                                                 TRAP    C$ESCAPE
 040246' 000340                                                                                 .WORD   L10146-.
 040250' 012737  000176  000332' 10$:         MOV     #2*SECOND,METER        ;WAIT FOR THE MICROMONITOR
 040256' 004737  017316'                      JSR     PC,CHKDNI
 040262' 103022                               BCC     20$                    ;OK
 040264' 012737  001000' 000310'              MOV     #SDNI,BITNAM
 040272' 012737  001277' 000312'              MOV     #SNSET,BITSTA
 040300' 012737  001342' 000314'              MOV     #SAFTER,PWHEN
 040306' 012737  001357' 000316'              MOV     #SGTCMD,PCOMND
 040314'                                      ERRHRD  153,ODDBYT,MSG1
 040314' 104456                                                                                 TRAP    C$ERHRD
 040316' 000231                                                                                 .WORD   153
 040320' 002765'                                                                                .WORD   ODDBYT
 040322' 012716'                                                                                .WORD   MSG1
```

```
040324'                                     ESCAPE  TST
040324' 104410                                                                          TRAP    C$ESCAPE
040326' 000260                                                                          .WORD   L10146-.
040330' 004737  017362'         20$:        JSR     PC.CLRDNI               ;CLEAR DNI BIT
040334' 103006                              BCC     25$
040336'                                     ERRHRD  154,ODDBYT,RACMG7       ;DNI DID NOT CLEAR!
040336' 104456                                                                          TRAP    C$ERHRD
040340' 000232                                                                          .WORD   154
040342' 002765'                                                                         .WORD   ODDBYT
040344' 012670'                                                                         .WORD   RACMG7
040346'                                     ESCAPE TST
040346' 104410                                                                          TRAP    C$ESCAPE
040350' 000236                                                                          .WORD   L10146-.
040352'                         25$:
040352'                           ENDSEG
040352'                                                                         10000$:
040352' 104405                                                                          TRAP    C$ESEG
                                ;
                                ;LOAD PCBB+0 WITH THE MINIMUM ODD BYTE COUNT
                                ;
040354' 012737  000101  000606'             MOV     #MINBYT+1,PCBB          ;BEGIN WITH MINIMUM ODD BYTE COUNT
040362'                         30$:
                                ;
                                ;CLEAR PCBB+2 AND PCBB+4. WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON'
                                ;STATE. EXECUTE MICROTEST #7 BY LOADING THE COMMAND FIELD OF PCSRO WITH A
                                ;'7'. WAIT FOR 'DNI'
                                ;
040362'                           BGNSEG
040362' 104404                                                                          TRAP    C$BSEG
040364' 005037  000610'                     CLR     PCBB+2                  ;THIS IS WHERE MICROCODE WILL PUT...
                                                                            ;RECEIVER BYTE COUNT
040370' 005037  000612'                     CLR     PCBB+4                  ;HERE IS WHERE MICROCODE WILL PUT...
                                                                            ;ACTUAL NUMBER OF BYTES RECEIVED
040374' 004737  020060'                     JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
040400' 103006                              BCC     35$                     ;OK
040402'                                     ERRHRD  156,ODDBYT,MSG46        ;PRINT ERROR
040402' 104456                                                                          TRAP    C$ERHRD
040404' 000234                                                                          .WORD   156
040406' 002765'                                                                         .WORD   ODDBYT
040410' 016666'                                                                         .WORD   MSG46
040412'                                     ESCAPE  TST                     ;LEAVE TEST
040412' 104410                                                                          TRAP    C$ESCAPE
040414' 000172                                                                          .WORD   L10146-.
040416' 012777  000007  137712  35$:        MOV     #7,@PCSRO               ;TELL T11 TO EXECUTE MICROTEST #7
040424' 012737  000176  000332'             MOV     #2*SECOND,METER         ;WAIT FOR DNI
040432' 004737  017316'                     JSR     PC,CHKDNI
040436' 103021                              BCC     40$
040440' 004737  020132'                     JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
040444' 103006                              BCC     36$                     ;NO, OK
040446'                                     ERRHRD  157,ODDBYT,MSG44        ;PRINT ERROR MESSAGE
040446' 104456                                                                          TRAP    C$ERHRD
040450' 000235                                                                          .WORD   157
040452' 002765'                                                                         .WORD   ODDBYT
040454' 016442'                                                                         .WORD   MSG44
040456'                                     ESCAPE  TST
040456' 104410                                                                          TRAP    C$ESCAPE
```

```
040460' 000126                                                                           .WORD   L10146-.
040462' 012702  000007          36$:    MOV     #7,R2                   ;MICROTEST #7 IS HUNG
040466'                                 ERRHRD  158,ODDBYT,MSG12
040466' 104456                                                                           TRAP    C$ERHRD
040470' 000236                                                                           .WORD   158
040472' 002765'                                                                          .WORD   ODDBYT
040474' 013466'                                                                          .WORD   MSG12
040476'                                 ESCAPE  TST
040476' 104410                                                                           TRAP    C$ESCAPE
040500' 000106                                                                           .WORD   L10146-.
                                ;
                                ;CHECK PCBB+2 TO SEE IF IT IS THE SAME AS PCBB+0, THIS WILL VERIFY THE
                                ;THE TRANSMIT AND RECEIVE BYTE COUNTS ARE THE SAME. THEN CHECK PCBB+4
                                ;TO SEE IF IT IS ONE LARGER THAN THE RECEIVE BYTE COUNT. THIS IS BECAUSE
                                ;THE RECEIVE CAN ONLY ADDRESS LINK MEMORY IN WORD BOUNDARIES SO THE ACTUAL
                                ;NUMBER OF BYTES THAT ARE CHANGED IS ONE MORE THAN WHAT WAS TRANSMITTED
                                ;
040502' 013701  000606'         40$:    MOV     PCBB,R1                 ;GET TRANSMIT BYTE COUNT
040506' 010102                          MOV     R1,R2
040510' 013703  000610'                 MOV     PCBB+2,R3               ;GET RECEIVE BYTE COUNT
040514' 013704  000612'                 MOV     PCBB+4,R4               ;GET ACTUAL NUMBER OF BYTES TRANSMITTED
040520' 020203                          CMP     R2,R3                   ;IS THE RECEIVE BYTE COUNT CORRECT?
040522' 001004                          BNE     45$                     ;NO, ERROR
040524' 162704  000001                  SUB     #1,R4                   ;ACCOUNT FOR THE GARBAGE BYTE
040530' 020104                          CMP     R1,R4                   ;IS THE ACTUAL NUMBER OF BYTES CORRECT?
040532' 001404                          BEQ     50$                     ;YES
040534'                         45$:    ERRHRD  159,ODDBYT,MSG19        ;PRINT ERROR MESSAGE
040534' 104456                                                                           TRAP    C$ERHRD
040536' 000237                                                                           .WORD   159
040540' 002765'                                                                          .WORD   ODDBYT
040542' 014052'                                                                          .WORD   MSG19
040544'                         50$:
040544'                           ENDSEG
040544'                                                                          10001$:
040544' 104405                                                                           TRAP    C$ESEG
                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
                                ;
040546' 004737  017362'                 JSR     PC,CLRDNI               ;CLEAR DNI BIT
040552' 103006                          BCC     60$
040554'                                 ERRHRD  160,ODDBYT,RACMG7       ;ERROR DNI DID NOT CLEAR!
040554' 104456                                                                           TRAP    C$ERHRD
040556' 000240                                                                           .WORD   160
040560' 002765'                                                                          .WORD   ODDBYT
040562' 012670'                                                                          .WORD   RACMG7
040564'                                 ESCAPE  TST
040564' 104410                                                                           TRAP    C$ESCAPE
040566' 000020                                                                           .WORD   L10146-.
                                ;
                                ;REPEAT MICROTEST #7 EACH TIME GIVING IT A LARGER ODD BYTE COUNT. CONTINUE
                                ;UNTIL THE MAXIMUM BYTE COUNT IS REACHED
                                ;
040570' 062737  000002  000606' 60$:    ADD     #2,PCBB                 ;UP THE BYTE COUNT
040576' 023727  000606' 002756          CMP     PCBB,#MAXBYT            ;HAVE WE DONE ALL ODD BYTE COUNTS?
040604' 003666                          BLE     30$                     ;NO, CONTINUE TEST WITH NEW BYTE COUNT
040606'                         ENDTST
```

```
 9349  040606'                    L10146:
 9350  040606' 104401                        TRAP    C$ETST
```

```
                                        .SBTTL  TEST 29: LINK MAXIMUM BYTE COUNTER TEST

                                ;*******************************************************************************
                                ;
                                ;THE RECEIVE BYTE CUUNTER IS A 12 BIT BINARY COUNTER THAT COUNTS THE NUMBER OF
                                ;BYTES THAT WERE RECEIVED DURING A DATAGRAM TRANSMISSION. THE BYTE COUNTER IS
                                ;INCREMENTED AS EACH BYTE IS RECEIVED. THE RECEIVE BYTE COUNTER HAS LOGIC THAT
                                ;DISABLES THE COUNTER IF THE MAXIMUM VALUE IS REACHED AND PREVENTS THE COUNTER
                                ;FROM ROLLING OVER TO ZERO.
                                ;
                                ;THIS TEST WILL CHECK THAT THE COUNTER 'TOPS OUT' AT THE MAXIMUM COUNTER VALUE.
                                ;IT DO THIS MICROMODULE 'D' MICROTEST #9 IS USED. IT WILL TRANSMIT A DATAGRAM
                                ;OF MAXIMUM COUNTER LENGTH OVER THE LOOPBACK. THE LINK CRC HARDWARE WILL BE
                                ;ALLOCATED TO THE TRANSMIT SIDE SO THAT CRC BYTES WILL APPENDED TO THE DATAGRAM.
                                ;THE LENGTH OF THE DATAGRAM WILL THEREFORE EXCEED THE LENGTH OF THE RECEIVE
                                ;BYTE COUNTER. THE RECEIVE COUNTER WILL BE CHECKED TO INSURE THAT THE COUNTER
                                ;HAS REMAINED AT THE MAXIMUM VALUE, IF NOT AN ERROR IS PASSED TO THE HOST.

                                ;TEST SEQUENCE:
                                ;       1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO
                                ;       2-VERIFY MICROMONITOR IS IN THE 'INMON' STATE
                                ;       3-SELECT MICROTEST #9
                                ;       4-WAIT FOR 'DNI'
                                ;       5-CHECK FOR AN ERROR IN PCSR1
                                ;       6-WRITE '1' TO CLEAR 'DNI'
                                ;
                                ;*******************************************************************************

040610'                         BGNTST
040610'                                                                 T29::
                                ;
                                ;CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                ;
040610'                          BGNSEG
040610' 104404                                                                  TRAP    C$BSEG
040612' 022737  000104  000326'         CMP     #'D,MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED?
040620' 001004                          BNE     5$                      ;NO
040622' 122777  000001  137510          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
040630' 001440                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
040632' 012737  000104  000326' 5$:     MOV     #'D,MICRO               ;GO LOAD MICROCODE MODULE D
040640' 004737  020340'                 JSR     PC,LODMIC
040644' 103002                          BCC     10$                     ;OK
040646'                                 ESCAPE  TST
040646' 104410                                                                  TRAP    C$ESCAPE
040650' 000256                                                                  .WORD   L10147-.
040652' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
040660' 004737  017316'                 JSR     PC,CHKDNI
040664' 103022                          BCC     20$                     ;OK
040666' 012737  001000' 000310'         MOV     #$DNI,BITNAM
040674' 012737  001277' 000312'         MOV     #$NSET,BITSTA
040702' 012737  001342' 000314'         MOV     #$AFTER,PWHEN
040710' 012737  001357' 000316'         MOV     #$GTCMD,PCOMND
040716'                                 ERRHRD  161,MAXCNT,MSG1 ;PRINT ERROR MESSAGE
040716' 104456                                                                  TRAP    C$ERHRD
```

```
040720' 000241                                                                   .WORD   161
040722' 003027'                                                                  .WORD   MAXCNT
040724' 012716'                                                                  .WORD   MSG1
040726'                                  ESCAPE  TST
040726' 104410                                                                   TRAP    C$ESCAPE
040730' 000176                                                                   .WORD   L10147-.
040732' 004737  017362'         20$:     JSR     PC,CLRDNI          ;CLEAR DNI
040736' 103006                           BCC     25$
040740'                                  ERRHRD  162,MAXCNT,RACMG7
040740' 104456                                                                   TRAP    C$ERHRD
040742' 000242                                                                   .WORD   162
040744' 003027'                                                                  .WORD   MAXCNT
040746' 012670'                                                                  .WORD   RACMG7
040750'                                  ESCAPE  TST
040750' 104410                                                                   TRAP    C$ESCAPE
040752' 000154                                                                   .WORD   L10147-.
040754'                         25$:
040754'                           ENDSEG
040754'                                                               10000$:
040754' 104405                                                                   TRAP    C$ESEG
                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'IN MONITOR' STATE,  LOAD COMMAND
                                ;FIELD OF PCSR0 WITH A 9 CAUSING THE MICROMONITOR TO EXECUTE MICROTEST #9.
                                ;WAIT FOR DNI
                                ;
040756'                           BGNSEG
040756' 104404                                                                   TRAP    C$BSEG
040760' 004737  020060'                  JSR     PC,CHKMON          ;WAIT FOR MICROMONITOR
040764' 103006                           BCC     30$                ;OK
040766'                                  ERRHRD  163,MAXCNT,MSG46   ;PRINT ERROR
040766' 104456                                                                   TRAP    C$ERHRD
040770' 000243                                                                   .WORD   163
040772' 003027'                                                                  .WORD   MAXCNT
040774' 016666'                                                                  .WORD   MSG46
040776'                                  ESCAPE  TST                ;LEAVE TEST
040776' 104410                                                                   TRAP    C$ESCAPE
041000' 000126                                                                   .WORD   L10147-.
041002' 012777  000011  137326  30$:     MOV     #9.,@PCSR0         ;TELL MICROMONITOR TO EXECUTE...
                                                                    ;MICROTEST #9
041010' 012737  000176  000332'          MOV     #2*SECOND,METER    ;PUT SOME TIME ON THE METER
041016' 004737  017316'                  JSR     PC,CHKDNI          ;WAIT FOR MICROTEST TO FINISH
041022' 103021                           BCC     40$                ;OK, IT FINISHED
041024' 004737  020132'                  JSR     PC,CHKINT          ;SEE IF ANY ERROR INTERRUPTS OCCURRED
041030' 103006                           BCC     35$                ;NO, OK
041032'                                  ERRHRD  164,MAXCNT,MSG44   ;PRINT ERROR MESSAGE
041032' 104456                                                                   TRAP    C$ERHRD
041034' 000244                                                                   .WORD   164
041036' 003027'                                                                  .WORD   MAXCNT
041040' 016442'                                                                  .WORD   MSG44
041042'                                  ESCAPE  TST
041042' 104410                                                                   TRAP    C$ESCAPE
041044' 000062                                                                   .WORD   L10147-.
041046' 012702  000011          35$:     MOV     #9.,R2             ;MICROTEST #
041052'                                  ERRHRD  165,MAXCNT,MSG12   ;TELL MICROTEST HUNG
041052' 104456                                                                   TRAP    C$ERHRD
041054' 000245                                                                   .WORD   165
```

```
041056' 003027'                                                                   .WORD   MAXCNT
041060' 013466'                                                                   .WORD   MSG12
041062'                              ESCAPE  TST
041062' 104410                                                                    TRAP    C$ESCAPE
041064' 000042                                                                    .WORD   L10147-.
                           ;
                           ;MICROTEST IS COMPLETE, NOW CHECK TO SEE IF IT WAS SUCCESSFULL
                           ;
041066' 122777 000003 137244 40$:    CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
041074' 001004                       BNE     50$                     ;NO
041076'                              ERRHRD  166,MAXCNT,MSG38        ;PRINT ERROR MESSAGE
041076' 104456                                                                    TRAP    C$ERHRD
041100' 000246                                                                    .WORD   166
041102' 003027'                                                                   .WORD   MAXCNT
041104' 016034'                                                                   .WORD   MSG38
                           ;
                           ;WRITE ONE TO CLEAR THE DNI BIT
                           ;
041106' 004737 017362'     50$:      JSR     PC,CLRDNI               ;CLEAR DNI
041112' 103004                       BCC     55$
041114'                              ERRHRD  167,MAXCNT,RACMG7
041114' 104456                                                                    TRAP    C$ERHRD
041116' 000247                                                                    .WORD   167
041120' 003027'                                                                   .WORD   MAXCNT
041122' 012670'                                                                   .WORD   RACMG7
041124'                    55$:
041124'                      ENDSEG
041124'                                                                   10001$:
041124' 104405                                                                    TRAP    C$ESEG
041126'                    ENDTST
041126'                                                                   L10147:
041126' 104401                                                                    TRAP    C$ETST
```

```
                        .SBTTL  TEST 30: FIFO TEST

                        ;*************************************************************************
                        ;
                        ;THERE ARE TWO FIFO'S USED IN THE DEUNA TO KEEP TRACK OF RECEIVER BUFFERS.
                        ;THE FIRST IS CALLED THE 'RECEIVER BUFFER AVAILABLE FIFO' AND THE SECOND IS
                        ;CALLED THE 'RECEIVER BUFFER DONE FIFO'.
                        ;
                        ;THE T11 LOADS THE RECEIVER BUFFER AVAILABLE FIFO WITH A LIST OF UNUSED 1K
                        ;BUFFERS IN LINK MEMORY. WHEN THE DEUNA SENSES THAT A PACKET IS COMING IN IT
                        ;PULLS AN AVAILABLE BUFFER ADDRESS FROM THE OUTPUT OF THE RECEIVER BUFFER
                        ;AVAILABLE FIFO AND USES IT TO ADDRESS LINK MEMORY FOR THE STORAGE OF THE
                        ;RECEIVED DATA. AFTER THE DATA HAS BEEN LOADED THE RECEIVER STATE MACHINE
                        ;PUTS THE USED BUFFER ADDRESS INTO THE RECEIVER BUFFER DONE FIFO WHERE AN
                        ;INTERRUPT IS GENERATED TO THE T11 WHEN IT BUBBLES TO THE TOP OF THE FIFO.
                        ;
                        ;THESE FIFO'S ARE 64 DEEP BY 4 BITS WIDE. THE OPERATIONAL MICROCODE ONLY FILLS
                        ;THE FIFO TO A MAXIMUM OF 16. THE 4 BIT WIDTH REPRESENTS BITS 14-11 OF THE
                        ;LINK MEMORY ADDRESS. THESE BITS ALLOW THE ADDRESSING OF A 1K BUFFER IN LINK
                        ;MEMORY.
                        ;
                        ;THIS TEST WILL VERIFY THAT THE RECEIVE BUFFER AVAILABLE FIFO AND THE RECEIVER
                        ;BUFFER DONE FIFO OPERATE CORRECTLY. THIS WILL BE DONE BY LOADING THE RECEIVER
                        ;BUFFER AVAILABLE FIFO WITH A 1K BUFFER ADDRESS THEN A PACKET WILL BE
                        ;TRANSMITTED IN LOOPBACK MODE. AFTER THE RECEIVER INTERRUPT OCCURS THE RECEIVER
                        ;BUFFER DONE FIFO IS READ AND THE ADDRESS IS COMPARED WITH WHAT WAS GIVEN
                        ;THE RECEIVER BUFFER AVAILABLE FIFO. THEY SHOULD BE THE SAME IF EVERYTHING IS
                        ;WORKING CORRECTLY. THE OPERATION IS PERFORMED WITH THE TRANSMITTER BUFFER
                        ;SET TO 0 AND WILL BE REPEATED WITH RECEIVER BUFFERS 1-15.
                        ;THIS TEST WILL USE MICROMODULE 'D' MICROTEST #10.
                        ;
                        ;PARAMETERS PASSED TO THE MICROCODE WILL BE FORMATTED IN THE PCBB AS FOLLOWS:
                        ;
                        ;PCBB+0: RECEIVE BUFFER ADDRESS
                        ;PCBB+2: RECEIVE BUFFER COMPLETED ADDRESS
                        ;
                        ;TEST SEQUENCE:
                        ;       1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO
                        ;       2-LOAD PCBB+0 WITH A RECEIVER BUFFER
                        ;       3-CLEAR PCBB+2
                        ;       4-VERIFY MICROMONITOR IS IN THE 'INMON' STATE
                        ;       5-SELECT MICROTEST #10
                        ;       6-WAIT FOR 'DNI' BIT
                        ;       7-VERIFY PCBB+2 SAME AS PCBB+0
                        ;       8-WRITE '1' TO CLEAR 'DNI'
                        ;       9-REPEAT STEPS 2-8 FOR RECEIVER BUFFERS 1-15
                        ;
                        ;*************************************************************************

 041130'                BGNTST
 041130'                                                                T30::
                        ;
                        ;CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO
                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                        ;
```

```
041130'                                   BGNSEG
041130' 104404                                                                              TRAP    C$BSEG
041132' 022737 000104 000326'                     CMP     #'D,MICRO          ;HAS MICROCODE MODULE 'D' BEEN LOADED
041140' 001004                                    BNE     5$                 ;NO
041142' 122777 000001 137170                      CMPB    #INMON,@PCSR1      ;YES, IS THE MICROMONITOR ACTIVE?
041150' 001440                                    BEQ     20$                ;YES SKIP LOADING THE MICROMODULE
041152' 012737 000104 000326'  5$:                MOV     #'D,MICRO          ;GO LOAD MICRO MODULE 'D'
041160' 004737 020340'                            JSR     PC,LODMIC
041164' 103002                                    BCC     10$                ;OK
041166'                                           ESCAPE  TST
041166' 104410                                                                              TRAP    C$ESCAPE
041170' 000326                                                                              .WORD   L10150-.
041172' 012737 000176 000332'  10$:               MOV     #2*SECOND,METER    ;WAIT FOR THE MICROMONITOR
041200' 004737 017316'                            JSR     PC,CHKDNI
041204' 103022                                    BCC     20$                ;OK
041206' 012737 001000' 000310'                    MOV     #$DNI,BITNAM
041214' 012737 001277' 000312'                    MOV     #$NSET,BITSTA
041222' 012737 001342' 000314'                    MOV     #$AFTER,PWHEN
041230' 012737 001357' 000316'                    MOV     #$GTCMD,PCOMND
041236'                                           ERRHRD  168,FIFTST,MSG1
041236' 104456                                                                              TRAP    C$ERHRD
041240' 000250                                                                              .WORD   168
041242' 003073'                                                                             .WORD   FIFTST
041244' 012716'                                                                             .WORD   MSG1
041246'                                           ESCAPE  TST
041246' 104410                                                                              TRAP    C$ESCAPE
041250' 000246                                                                              .WORD   L10150-.
041252' 004737 017362'         20$:               JSR     PC,CLRDNI          ;CLEAR DNI BIT
041256' 103006                                    BCC     25$
041260'                                           ERRHRD  169,FIFTST,RACMG7  ;DNI DID NOT CLEAR!
041260' 104456                                                                              TRAP    C$ERHRD
041262' 000251                                                                              .WORD   169
041264' 003073'                                                                             .WORD   FIFTST
041266' 012670'                                                                             .WORD   RACMG7
041270'                                           ESCAPE TST
041270' 104410                                                                              TRAP    C$ESCAPE
041272' 000224                                                                              .WORD   L10150-.
041274'                        25$:
041274'                          ENDSEG
041274'                                                                            10000$:
041274' 104405                                                                              TRAP    C$ESEG
                               ;
                               ;LINK MEMORY STARTS AT MICROMEMORY ADDRESS 100000, THE MICROTEST WILL USE
                               ;100000 AS THE TRANSMIT BUFFER ADDRESS SO THE FIRST RECEIVE BUFFER ADDRESS
                               ;WE WILL PASS TO THE MICROCODE WILL BE 104000
                               ;
041276' 012701 104000                             MOV     #LINADR+SIZ1K,R1   ;FIRST RECEIVE BUFFER STARTS 1K FROM
                                                                             ;BASE OF LINK MEMORY
041302'                        30$:
                               ;
                               ;PUT THE BUFFER ADDRESS WE WANT THE MICROCODE TO LOAD INTO THE AVAILABLE BUFFER
                               ;INTO PCBB AND CLEAR PCBB+2. WAIT FOR THE MICROMONITOR TO BECOME READY, THEN
                               ;SELECT MICROTEST #10 BY LOADING THE COMMAND FIELD OF PCSR0 WITH A 10. WAIT FOR
                               ;'DNI'
                               ;
041302'                          BGNSEG
```

```
041302' 104404                                                                          TRAP    C$BSEG
041304' 010137  000606'                  MOV     R1,PCBB                 ;PASS AVAILABLE RECEIVER BUFFER TO DEUNA
041310' 005037  000610'                  CLR     PCBB+2                  ;HERE IS WHERE THE MICRO WILL PUT THE
                                                                         ;'DONE' RECEIVE BUFFER FIFO ADDRESS
041314' 004737  020060'                  JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
041320' 103006                           BCC     35$                     ;OK
041322'                                  ERRHRD  170,FIFTST,MSG46        ;PRINT ERROR
041322' 104456                                                                          TRAP    C$ERHRD
041324' 000252                                                                          .WORD   170
041326' 003073'                                                                         .WORD   FIFTST
041330' 016666'                                                                         .WORD   MSG46
041332'                                  ESCAPE  TST                     ;LEAVE TEST
041332' 104410                                                                          TRAP    C$ESCAPE
041334' 000162                                                                          .WORD   L10150-.
041336' 012777  000012  136772  35$:     MOV     #10.,@PCSR0             ;TELL T11 TO EXECUTE MICROTEST #10
041344' 012737  000176  000332'          MOV     #2*SECOND,METER         ;WAIT FOR DNI
041352' 004737  017316'                  JSR     PC,CHKDNI
041356' 103021                           BCC     45$                     ;OK
041360' 004737  020132'                  JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
041364' 103006                           BCC     40$                     ;NO, OK
041366'                                  ERRHRD  171,FIFTST,MSG44        ;PRINT ERROR MESSAGE
041366' 104456                                                                          TRAP    C$ERHRD
041370' 000253                                                                          .WORD   171
041372' 003073'                                                                         .WORD   FIFTST
041374' 016442'                                                                         .WORD   MSG44
041376'                                  ESCAPE  TST
041376' 104410                                                                          TRAP    C$ESCAPE
041400' 000116                                                                          .WORD   L10150-.
041402' 012702  000012          40$:     MOV     #10.,R2                 ;MICROTEST #10 IS HUNG
041406'                                  ERRHRD  172,FIFTST,MSG12        ;PRINT ERROR MESSAGE
041406' 104456                                                                          TRAP    C$ERHRD
041410' 000254                                                                          .WORD   172
041412' 003073'                                                                         .WORD   FIFTST
041414' 013466'                                                                         .WORD   MSG12
041416'                                  ESCAPE TST
041416' 104410                                                                          TRAP    C$ESCAPE
041420' 000076                                                                          .WORD   L10150-.
                                ;
                                ;OK, THE MICROCODE HAS FILLED PCBB+2 WITH THE OUTPUT OF THE RECEIVER BUFFER
                                ;DONE FIFO HOWEVER, THIS VALUE IS NOT IN ADDRESS FORM BECAUSE THERE ARE SOME
                                ;UNUSED BITS (15,10:00) THAT COULD BE FLOATING SO, STRIP THESE BITS FIRST.
                                ;THEN SET BIT 15 TO FORCE THE ADDRESS TO BE A LINK MEMORY ADDRESS. COMPARE
                                ;THE ADDRESS GIVEN TO THE MICROCODE WITH THE ONE JUST GENERATED, THEY SHOULD
                                ;BE IDENTICAL, IF NOT, PRINT ERROR.
                                ;
041422' 042737  103777  000610' 45$:     BIC     #103777,PCBB+2          ;STRIP OFF THE FLOATING BITS
041430' 052737  100000  000610'          BIS     #100000,PCBB+2          ;MAKE IT A LINK MEMORY ADDRESS
041436' 023737  000606' 000610'          CMP     PCBB,PCBB+2             ;DID THE DEUNA USE THE BUFFER AVAILABLE
041444' 001404                           BEQ     50$                     ;YES
041446'                                  ERRHRD  173,FIFTST,MSG35        ;NO, PRINT ERROR MESSAGE
041446' 104456                                                                          TRAP    C$ERHRD
041450' 000255                                                                          .WORD   173
041452' 003073'                                                                         .WORD   FIFTST
041454' 015672'                                                                         .WORD   MSG35
041456'                         50$:
041456'                                  ENDSEG
```

```
041456'                                                                    10001$:
041456' 104405                                                                     TRAP    C$ESEG

                                ;
                                ;WRITE '1' TO CLEAR 'DNI' BIT
                                ;
041460' 004737  017362'                 JSR     PC,CLRDNI               ;GO CLEAR DNI
041464' 103006                          BCC     60$                     ;OK
041466'                                 ERRHRD  174,FIFTST,RACMG7       ;ERROR DNI DID NOT CLEAR!
041466' 104456                                                                     TRAP    C$ERHRD
041470' 000256                                                                     .WORD   174
041472' 003073'                                                                    .WORD   FIFTST
041474' 012670'                                                                    .WORD   RACMG7
041476'                                 ESCAPE TST
041476' 104410                                                                     TRAP    C$ESCAPE
041500' 000016                                                                     .WORD   L10150-.
                                ;
                                ;CHECK TO SEE IF WE HAVE TRIED ALL 15 RECEIVER BUFFER ADDRESSES, IF NOT,
                                ;GENERATE A NEW ADDRESS TO PASS TO THE MICROCODE AND RUN THRU THE TEST AGAIN
                                ;
041502' 020127  174000          60$:    CMP     R1,#LINADR+<SIZ1K*15.>  ;HAVE WE TRIED ALL 15 BUFFERS
041506' 001403                          BEQ     70$                     ;YES ALL DONE
041510' 062701  004000                  ADD     #SIZ1K,R1               ;NO POINT TO NEXT RECEIVER BUFFER
041514' 000672                          BR      30$                     ;DO AGAIN
041516'                         70$:
041516'                         ENDTST
041516'                                                                    L10150:
041516' 104401                                                                     TRAP    C$ETST
```

```
                                          .SBTTL  TEST 31: RECEIVER LINK MEMORY ADDRESS TEST

                                          ;*******************************************************************************
                                          ;
                                          ;THIS TEST WILL VERIFY THAT BUFFERS 1-15 OF LINK MEMORY CAN BE ADDRESSED
                                          ;CORRECTLY BY THE RECEIVER. THIS WILL BE DONE BY DIRECTING THE MICROCODE TO
                                          ;TRANSMIT A DATA PATTERN FROM BUFFER 0 AND TO RECEIVE THE DATA IN BUFFER X
                                          ;WHERE X = 1-15. THEN A CHECK WILL BE MADE TO SEE IF THE PATTERN NOT ONLY
                                          ;ARRIVED IN THE CORRECT RECEIVER BUFFER BUT THAT THE PATTERN DOES NOT SHOW UP
                                          ;ANYWHERE ELSE IN LINK MEMORY EXCEPT WHERE IT WAS SUPPOSE TO.
                                          ;
                                          ;THIS TEST WILL USE MICROMODULE 'D' MICROTEST #11. THIS MICROTEST ACCEPTS 2
                                          ;PARAMETERS: THE TRANSMIT BUFFER AND THE RECEIVER BUFFER. IT WILL SET UP
                                          ;A DATA PATTERN IN THE TRANSMIT BUFFER AND TELL THE LINK TO TRANSMIT, IN
                                          ;LOOPBACK MODE, FROM THE TRANSMIT BUFFER GIVEN TO THE RECEIVER BUFFER GIVEN.
                                          ;AFTER THE RECEIVER INTERRUPT, THE DATA IS CHECKED IN THE EXPECTED RECEIVER
                                          ;BUFFER FOR THE CORRECT DATA PATTERN. THEN ALL OF LINK MEMORY (EXCEPT FOR THE
                                          ;TRANSMIT BUFFER) IS CHECKED TO SEE IF THE PATTERN ENDS UP ELSEWHERE. IF
                                          ;AN ERROR IS FOUND THE MICROCODE PASSES THE ADDRESS OF LINK MEMORY WHERE THE
                                          ;ERROR WAS FOUND, THE DATA THAT WAS FOUND THERE ALONG WITH THE DATA THAT SHOULD
                                          ;HAVE BEEN THERE.
                                          ;
                                          ;THE PARAMETERS FOR THE MICROCODE ARE FORMATED IN THE PCBB AS FOLLOWS:
                                          ;
                                          ;PCBB+0: RECEIVER BUFFER ADDRESS
                                          ;PCBB+2: TRANSMIT BUFFER ADDRESS
                                          ;PCBB+4: LINK MEMORY ADDRESS (IF ERROR)
                                          ;PCBB+6: GOOD DATA PATTERN (IF ERROR)
                                          ;PCBB+10: BAD DATA PATTERN (IF ERROR)
                                          ;
                                          ;TEST SEQUENCE:
                                          ;       1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO
                                          ;       2-SET PCBB+0 WITH TRANSMIT BUFFER 0 ADDRESS
                                          ;       3-SET PCBB+2 WITH RECEIVER BUFFER ADDRESS
                                          ;       4-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                          ;       5-SELECT MICROTEST #11
                                          ;       6-WAIT FOR 'DNI'
                                          ;       7-CHECK FOR ERROR IN PCSR1 IF SO PRINT ERROR MESSAGE
                                          ;       8-WRITE '1' TO CLEAR 'DNI'
                                          ;       9-REPEAT STEPS 2-8 WITH RECEIVER BUFFERS 1-15
                                          ;
                                          ;*******************************************************************************

 041520'                                  BGNTST
 041520'                                                                                T31::

                                          ;
                                          ;CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                          ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                          ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                          ;
 041520'                                    BGNSEG
 041520' 104404                                                                         TRAP    C$BSEG
 041522' 022737  000104  000326'                  CMP     #'D,MICRO           ;HAS MICROCODE MODULE 'D' BEEN LOADED
 041530' 001004                                   BNE     5$                  ;NO
 041532' 122777  000001  136600                   CMPB    #INMON,@PCSR1       ;YES, IS THE MICROMONITOR ACTIVE?
 041540' 001440                                   BEQ     20$                 ;YES SKIP LOADING THE MICROMODULE
```

```
041542' 012737  000104  000326' 5$:     MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
041550' 004737  020340'                 JSR     PC,LODMIC
041554' 103002                          BCC     10$                     ;OK
041556'                                 ESCAPE  TST
041556' 104410                                                                  TRAP    C$ESCAPE
041560' 000322                                                                  .WORD   L10151-.
041562' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
041570' 004737  017316'                 JSR     PC,CHKDNI
041574' 103022                          BCC     20$                     ;OK
041576' 012737  001000' 000310'         MOV     #$DNI,BITNAM
041604' 012737  001277' 000312'         MOV     #$NSET,BITSTA
041612' 012737  001342' 000314'         MOV     #$AFTER,PWHEN
041620' 012737  001357' 000316'         MOV     #$GTCMD,PCOMND
041626'                                 ERRHRD  175,RLNKAD,MSG1
041626' 104456                                                                  TRAP    C$ERHRD
041630' 000257                                                                  .WORD   175
041632' 003114'                                                                 .WORD   RLNKAD
041634' 012716'                                                                 .WORD   MSG1
041636'                                 ESCAPE  TST
041636' 104410                                                                  TRAP    C$ESCAPE
041640' 000242                                                                  .WORD   L10151-.
041642' 004737  017362'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
041646' 103006                          BCC     25$
041650'                                 ERRHRD  176,RLNKAD,RACMG7       ;DNI DID NOT CLEAR!
041650' 104456                                                                  TRAP    C$ERHRD
041652' 000260                                                                  .WORD   176
041654' 003114'                                                                 .WORD   RLNKAD
041656' 012670'                                                                 .WORD   RACMG7
041660'                                 ESCAPE TST
041660' 104410                                                                  TRAP    C$ESCAPE
041662' 000220                                                                  .WORD   L10151-.
041664'                         25$:
041664'                           ENDSEG
041664'                                                                 10000$:
041664' 104405                                                                  TRAP    C$ESEG
                                ;
                                ;LOAD PCBB+2 WITH THE ADDRESS OF THE FIRST TRANSMIT BUFFER
                                ;LOAD PCBB+0 WITH THE FIRST RECEIVER BUFFER ADDRESS
                                ;
041666' 012701  002756                  MOV     #MAXBYT,R1              ;TRANSMIT BYTE COUNT
041672' 012737  100000  000610'         MOV     #LINADR,PCBB+2          ;SET TRANSMIT BUFFER AT BASE OF LINK
                                                                        ;MEMORY
041700' 012737  104000  000606'         MOV     #LINADR+SIZ1K,PCBB      ;FIRST RECEIVE BUFFER STARTS 1K FROM
                                                                        ;BASE OF LINK MEMORY
041706'                         30$:
041706'                           BGNSEG
041706' 104404                                                                  TRAP    C$BSEG
                                ;
                                ;WAIT FOR THE MICROMONITOR, THEN SELECT MICROTEST #11 BY LOADING THE COMMAND
                                ;FIELD OF PCSRO WITH 11 , WAIT FOR 'DNI'
                                ;
041710' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
041714' 103006                          BCC     35$                     ;OK
041716'                                 ERRHRD  177,RLNKAD,MSG46        ;PRINT ERROR
041716' 104456                                                                  TRAP    C$ERHRD
041720' 000261                                                                  .WORD   177
```

```
041722' 003114'                                                                                .WORD   RLNKAD
041724' 016666'                                                                                .WORD   MSG46
041726'                                 ESCAPE  TST                     ;LEAVE TEST
041726' 104410                                                                                 TRAP    C$ESCAPE
041730' 000152                                                                                 .WORD   L10151-.
041732' 012777  000013  136376  35$:    MOV     #11.,@PCSR0             ;TELL T11 TO EXECUTE MICROTEST #11
041740' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
041746' 004737  017316'                 JSR     PC,CHKDNI
041752' 103021                          BCC     45$                     ;OK
041754' 004737  020132'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
041760' 103006                          BCC     40$                     ;NO, OK
041762'                                 ERRHRD  178,RLNKAD,MSG44        ;PRINT ERROR MESSAGE
041762' 104456                                                                                 TRAP    C$ERHRD
041764' 000262                                                                                 .WORD   178
041766' 003114'                                                                                .WORD   RLNKAD
041770' 016442'                                                                                .WORD   MSG44
041772'                                 ESCAPE  TST
041772' 104410                                                                                 TRAP    C$ESCAPE
041774' 000106                                                                                 .WORD   L10151-.
041776' 012702  000013          40$:    MOV     #11.,R2                 ;MICROTEST #11 IS HUNG
042002'                                 ERRHRD  179,RLNKAD,MSG12        ;PRINT ERROR MESSAGE
042002' 104456                                                                                 TRAP    C$ERHRD
042004' 000263                                                                                 .WORD   179
042006' 003114'                                                                                .WORD   RLNKAD
042010' 013466'                                                                                .WORD   MSG12
042012'                                 ESCAPE TST
042012' 104410                                                                                 TRAP    C$ESCAPE
042014' 000066                                                                                 .WORD   L10151-.
                                ;
                                ;IF THE MICROCODE FOUND AN ERROR IT SETS PCSR1 TO THE ERROR STATE SO, CHECK
                                ;PCSR1 TO SEE IF AN ERROR OCCURRED
                                ;IF SO PRINT ERROR MESSAGE
                                ;
042016' 122777  000003  136314  45$:    CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
042024' 001004                          BNE     50$                     ;NO
042026'                                 ERRHRD  180,RLNKAD,MSG42        ;YES, PRINT ERROR MESSAGE
042026' 104456                                                                                 TRAP    C$ERHRD
042030' 000264                                                                                 .WORD   180
042032' 003114'                                                                                .WORD   RLNKAD
042034' 016250'                                                                                .WORD   MSG42
042036'                         50$:
042036'                           ENDSEG
042036'                                                                                10001$:
042036' 104405                                                                                 TRAP    C$ESEG
                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
                                ;
042040' 004737  017362'                 JSR     PC,CLRDNI               ;GO CLEAR DNI
042044' 103006                          BCC     60$                     ;OK
042046'                                 ERRHRD  181,RLNKAD,RACMG7       ;ERROR DNI DID NOT CLEAR!
042046' 104456                                                                                 TRAP    C$ERHRD
042050' 000265                                                                                 .WORD   181
042052' 003114'                                                                                .WORD   RLNKAD
042054' 012670'                                                                                .WORD   RACMG7
042056'                                 ESCAPE TST
042056' 104410                                                                                 TRAP    C$ESCAPE
```

```
042060' 000022                                                                  .WORD   L10151-.
                                        ;
                                        ;THIS TEST IS TO BE REPEATED 15 TIMES ONECE FOR EACH RECEIVER BUFFER (1-15).
                                        ;CHECK TO SEE IF WE HAVE PASSED THE LAST RECEIVER BUFFER TO THE MICROCODE.
                                        ;IF SO STOP, ELSE ADD 1K TO THE CONTENTS OF PCBB TO MAKE THE NEW RECEIVER
                                        ;BUFFER AND REPEAT THE TEST AGAIN
                                        ;
042062' 023727  000606' 174000  60$:    CMP     PCBB,#LINADR+<SIZ1K*15.> ;HAVE WE TRIED ALL 15 BUFFERS
042070' 001404                          BEQ     70$                      ;YES ALL DONE
042072' 062737  004000  000606'         ADD     #SIZ1K,PCBB              ;NO POINT TO NEXT RECEIVER BUFFER
042100' 000702                          BR      30$                      ;DO AGAIN
042102'                         70$:
042102'                         ENDTST
042102'                                                             L10151:
042102' 104401                                                              TRAP    C$ETST
```

```
                                            .SBTTL  TEST 32: TRANSMITTER LINK MEMORY ADDRESS TEST

                                            ;*****************************************************************
                                            ;
                                            ;THIS TEST WILL VERIFY THAT BUFFERS 1-15 CAN BE CORRECTLY ADDRESSED BY THE
                                            ;TRANSMITTER. THIS TEST IS IDENTICAL TO THE RECEIVER ADDRESS TEST IN THAT
                                            ;IT USES THE SAME MICROCODE (MICROMODULE 'D' MICROTEST #11) EXCEPT IT
                                            ;FIXES THE RECEIVER BUFFER AT 0 AND VARIES THE TRANSMIT BUFFER FROM 1-15.
                                            ;
                                            ;TEST SEQUENCE:
                                            ;       1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO
                                            ;       2-SET PCBB+0 TO 100000 (BASE OF LINK MEMORY) AS RECEIVER BUFFER ADDRESS
                                            ;       3-SET PCBB+2 WITH A TRANSMITTER BUFFER ADDRESS
                                            ;       4-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                            ;       5-SELECT MICROTEST #11
                                            ;       6-WAIT FOR 'DNI'
                                            ;       7-CHECK PCSR1 FOR AN ERROR CONDITION IF SO PRINT ERROR MESSAGE
                                            ;       8-WRITE '1' TO CLEAR 'DNI'
                                            ;       9-REPEAT STEPS 2-8 FOR ALL TRANSMIT BUFFERS 1-15
                                            ;
                                            ;*****************************************************************

 042104'                                    BGNTST
 042104'                                                                                T32::

                                            ;
                                            ;CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                            ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                            ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                            ;
 042104'                                      BGNSEG
 042104' 104404                                                                             TRAP    C$BSEG
 042106' 022737  000104  000326'       CMP     #'D,MICRO               ;HAS MICROCODE MODULE 'D' BEEN LOADED
 042114' 001004                        BNE     5$                      ;NO
 042116' 122777  000001  136214        CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
 042124' 001440                        BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
 042126' 012737  000104  000326' 5$:   MOV     #'D,MICRO               ;GO LOAD MICRO MODULE 'D'
 042134' 004737  020340'               JSR     PC,LODMIC
 042140' 103002                        BCC     10$                     ;OK
 042142'                               ESCAPE  TST
 042142' 104410                                                                             TRAP    C$ESCAPE
 042144' 000322                                                                             .WORD   L10152-.
 042146' 012737  000176  000332' 10$:  MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
 042154' 004737  017316'               JSR     PC,CHKDNI
 042160' 103022                        BCC     20$                     ;OK
 042162' 012737  001000' 000310'       MOV     #$DNI,BITNAM
 042170' 012737  001277' 000312'       MOV     #$NSET,BITSTA
 042176' 012737  001342' 000314'       MOV     #$AFTER,PWHEN
 042204' 012737  001357' 000316'       MOV     #$GTCMD,PCOMND
 042212'                               ERRHRD  182,TLNKAD,MSG1
 042212' 104456                                                                             TRAP    C$ERHRD
 042214' 000266                                                                             .WORD   182
 042216' 003165'                                                                            .WORD   TLNKAD
 042220' 012716'                                                                            .WORD   MSG1
 042222'                               ESCAPE  TST
 042222' 104410                                                                             TRAP    C$ESCAPE
 042224' 000242                                                                             .WORD   L10152-.
```

```
042226' 004737  017362'          20$:    JSR     PC,CLRDNI                 ;CLEAR DNI BIT
042232' 103006                           BCC     25$
042234'                                  ERRHRD  183,TLNKAD,RACMG7         ;DNI DID NOT CLEAR!
042234' 104456                                                                                     TRAP    C$ERHRD
042236' 000267                                                                                     .WORD   183
042240' 003165'                                                                                    .WORD   TLNKAD
042242' 012670'                                                                                    .WORD   RACMG7
042244'                                  ESCAPE TST
042244' 104410                                                                                     TRAP    C$ESCAPE
042246' 000220                                                                                     .WORD   L10152-.
042250'                          25$:
042250'                            ENDSEG
042250'                                                                                   10000$:
042250' 104405                                                                                     TRAP    C$ESEG
                                 ;
                                 ;LOAD PCBB+0 WITH THE BASE ADDRESS OF LINK MEMORY. THIS IS WHERE THE RECEIVER
                                 ;BUFFER WILL BE FIXED AT. LOAD PCBB+2 WITH THE ADDRESS OF BUFFER #1 (104000)
                                 ;
042252' 012701  002756                   MOV     #MAXBYT,R1                ;TRANSMIT BYTE COUNT
042256' 012737  100000  000606'          MOV     #LINADR,PCBB              ;SET RECEIVE BUFFER AT BASE OF LINK
                                                                           ;MEMORY
042264' 012737  104000  000610'          MOV     #LINADR+SIZ1K,PCBB+2      ;TRANSMIT BUFFER STARTS 1K FROM
                                                                           ;BASE OF LINK MEMORY
042272'                          30$:
                                 ;
                                 ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE. THEN EXECUTE MICROTEST
                                 ;#11 BY LOADING PCSRO COMMAND FIELD BITS WITH A 11. WAIT FOR 'DNI'
                                 ;
042272'                            BGNSEG
042272' 104404                                                                                     TRAP    C$BSEG
042274' 004737  020060'                  JSR     PC,CHKMON                 ;WAIT FOR MICROMONITOR
042300' 103006                           BCC     35$                       ;OK
042302'                                  ERRHRD  184,TLNKAD,MSG46          ;PRINT ERROR
042302' 104456                                                                                     TRAP    C$ERHRD
042304' 000270                                                                                     .WORD   184
042306' 003165'                                                                                    .WORD   TLNKAD
042310' 016666'                                                                                    .WORD   MSG46
042312'                                  ESCAPE  TST                       ;LEAVE TEST
042312' 104410                                                                                     TRAP    C$ESCAPE
042314' 000152                                                                                     .WORD   L10152-.
042316' 012777  000013  136012   35$:    MOV     #11.,@PCSRO               ;TELL T11 TO EXECUTE MICROTEST #11
042324' 012737  000176  000332'          MOV     #2*SECOND,METER           ;WAIT FOR DNI
042332' 004737  017316'                  JSR     PC,CHKDNI
042336' 103021                           BCC     45$                       ;OK
042340' 004737  020132'                  JSR     PC,CHKINT                 ;SEE IF ANY ERROR INTERRUPTS OCCURRED
042344' 103006                           BCC     40$                       ;NO, OK
042346'                                  ERRHRD  185,TLNKAD,MSG44          ;PRINT ERROR MESSAGE
042346' 104456                                                                                     TRAP    C$ERHRD
042350' 000271                                                                                     .WORD   185
042352' 003165'                                                                                    .WORD   TLNKAD
042354' 016442'                                                                                    .WORD   MSG44
042356'                                  ESCAPE  TST
042356' 104410                                                                                     TRAP    C$ESCAPE
042360' 000106                                                                                     .WORD   L10152-.
042362' 012702  000013           40$:    MOV     #11.,R2                   ;MICROTEST #11 IS HUNG
042366'                                  ERRHRD  186,TLNKAD,MSG12          ;PRINT ERROR MESSAGE
```

```
042366' 104456                                                          TRAP    C$ERHRD
042370' 000272                                                          .WORD   186
042372' 003165'                                                         .WORD   TLNKAD
042374' 013466'                                                         .WORD   MSG12
042376'                                    ESCAPE TST
042376' 104410                                                          TRAP    C$ESCAPE
042400' 000066                                                          .WORD   L10152-.
                                  ;
                                  ;THE MICROCODE WILL SET PCSR1 TO AN ERROR CONDITION IF IT DETECTED AN ERROR.
                                  ;SO CHECK PCSR1 IF ERROR, PRINT ERROR MESSAGE
                                  ;
042402' 122777  000003  135730    45$:     CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
042410' 001004                             BNE     50$                     ;NO
042412'                                    ERRHRD  187,TLNKAD,MSG42        ;YES, PRINT ERROR MESSAGE
042412' 104456                                                          TRAP    C$ERHRD
042414' 000273                                                          .WORD   187
042416' 003165'                                                         .WORD   TLNKAD
042420' 016250'                                                         .WORD   MSG42
042422'                           50$:
042422'                             ENDSEG
042422'                                                          10001$:
042422' 104405                                                          TRAP    C$ESEG
                                  ;
                                  ;WRITE '1' TO CLEAR 'DNI'
                                  ;
042424' 004737  017362'                    JSR     PC,CLRDNI               ;GO CLEAR DNI
042430' 103006                             BCC     60$                     ;OK
042432'                                    ERRHRD  188,TLNKAD,RACMG7       ;ERROR DNI DID NOT CLEAR!
042432' 104456                                                          TRAP    C$ERHRD
042434' 000274                                                          .WORD   188
042436' 003165'                                                         .WORD   TLNKAD
042440' 012670'                                                         .WORD   RACMG7
042442'                                    ESCAPE TST
042442' 104410                                                          TRAP    C$ESCAPE
042444' 000022                                                          .WORD   L10152-.
                                  ;
                                  ;THIS TEST IS TO BE REPEATED FOR EACH TRANSMIT BUFFER SO CHECK IF THE MICROCODE
                                  ;HAS BEEN PASSED THE LAST TRANSMIT BUFFER, IF NOT, CHANGE PCBB+2 TO POINT
                                  ;TO THE NEXT TRANSMIT BUFFER BY ADDING 1K TO IT
                                  ;
042446' 023727  000610' 174000    60$:     CMP     PCBB+2,#LINADR+<SIZ1K*15.> ;TRIED BUFFERS 1-15 FOR TRANSMITTER?
042454' 001404                             BEQ     70$                     ;YES ALL DONE
042456' 062737  004000  000610'            ADD     #SIZ1K,PCBB+2           ;NO POINT TO NEXT TRANSMIT BUFFER
042464' 000702                             BR      30$                     ;DO AGAIN
042466'                           70$:
042466'                           ENDTST
042466'                                                          L10152:
042466' 104401                                                          TRAP    C$ETST
```

```
.SBTTL  TEST 33: LINK MEMORY ARBITRATION TEST

;*******************************************************************************
;
;THE LINK MEMORY CAN BE ACCESSED BY FOUR PROCESSES; THE T-11 PROCESSOR, THE
;DMA ENGINE, THE RECEIVE STATE MACHINE AND THE TRANSMIT STATE MACHINE. THE
;PORT MODULE HAS ARBITRATION CIRCUITRY TO MANAGE LINK MEMORY ACCESSES. THIS
;CIRCUITRY PREVENTS CONFLICKS BETWEEN PROCESSES AND ASSURES THAT HIGHER
;PRIORITY PROCESSES GET PRECEDENCE.
;
;THIS TEST WILL VERIFY THE ABLILITY OF THE LINK MEMORY ARITRATOR TO HANDLE
;SIMULTANEOUS REQUESTS BY FOUR PROCESSES. EACH OF THESE PROCESSES WILL INVOLVE
;TASKS THAT REQUIRE HEAVY ACCESSES OF LINK MEMORY. DATA WILL BE MOVED INTO OR
;OUT OF LINK MEMORY BY EACH. WHEN THAT TASKS ARE FINISHED THE DATA WILL BE
;VERIFIED.
;
;THE FOUR PROCESSES ARE:
;
;       1-TRANSMIT STATE MACHINE
;
;               WILL TRANSMIT A DATAGRAM OF MAXIMUM DATA LENGTH IN LOOPBACK
;               MODE. THE DATA FIELD WILL CONTAIN A BIT PATTERN STRING OF
;               TWO 1'S FOLLOWED BY TWO 0'S I.E. 31463 (OCTAL).
;
;       2-RECEIVE STATE MACHINE
;
;               WILL RECEIVE A DATAGRAM OF MAXIMUM DATA LENGTH OVER THE
;               LOOPBACK. THE RECEIVE DATA BUFFER WILL BE FILLED WITH 0'S
;               PRIOR TO THE RECEPTION.
;
;       3-T-11 MICROPROCESSOR DMA
;
;               A 1K BUFFER IN LINK MEMORY WILL BE FILLED WITH AN ALL 1'S DATA
;               PATTERN PRIOR TO THE OPERATION THEN ALTERNATING 1'S AND 0'S
;               DATA PATTERN WILL BE WRITTEN.
;
;       4-DMA ENGINE
;
;               WILL TRANSFER A 1K BLOCK OF DATA FROM LINK MEMORY TO UNIBUS
;               MEMORY. THE DATA IN LINK MEMORY WILL A BIT PATTERN STRING
;               OF FOUR 1'S FOLLOWED BY A STRING OF FOUR 0'S. THE BUFFER
;               IN UNIBUS MEMORY WILL BE CLEARED PRIOR TO THE OPERATION.
;
;THE FOUR PROCESSES WILL WORK OUT OF FOUR SEPARATE AREAS OF LINK MEMORY.
;
;BASE OF LINK MEMORY--> +-------------------------+
;                       ! RECEIVE BUFFER          !
;       BASE + 4000---> +-------------------------+
;                       ! TRANSMIT BUFFER         !
;       BASE + 10000--> +-------------------------+
;                       ! DMA ENGINE BUFFER       !
;       BASE + 14000--> +-------------------------+
;                       ! T-11 PROCESSOR BUFFER   !
;                       +-------------------------+
;
;
;THIS WILL ALLOW THE ARITRATION CIRCUITRY TO BE TESTED AND YET ALLOWS THE DATA
```

```
                                                ;TO BE VERIFIED EASILY AND ASSOCIATED WITH A SINGLE PROCESS.
                                                ;
                                                ;A DATAGRAM WILL BE LOOPED BACK FROM THE TRANSMIT BUFFER TO THE RECEIVE BUFFER.
                                                ;AS THE DATAGRAM IS BEING TRANSFERRED, THE T-11 PROCESSOR WILL FILL IT'S BUFFER
                                                ;AND THE DMA ENGINE WILL TRANSFER IT'S BUFFER FROM LINK MEMORY TO UNIBUS MEMORY.
                                                ;
                                                ;WHEN THE RECEIVE STATE MACHINE IS DONE, THE T-11 WILL VERIFY THE DATA IN THE
                                                ;RECEIVE BUFFER. IF AN ERROR IS FOUND PCSR1 WILL BE SET TO INDICATE AN ERROR.
                                                ;
                                                ;THE HOST WILL WAIT FOR THE MICROCODE TO FINISH AND WHEN DONE, IT WILL VERIFY
                                                ;THE DATA TRANSFERRED BY THE DMA ENGINE TO UNIBUS MEMORY.
                                                ;
                                                ;THE FIRST 3 WORDS OF THE PCBB ARE USED FOR ERROR INFORMATION, THE REST
                                                ;WILL BE THE UNIBUS ADDRESS THAT THE DMA ENGINE WILL TRANSFER TO.
                                                ;
                                                ;                    +------------------------+
                                                ; PCBB+0:            !    EXPECTED PATTERN    !
                                                ;                    +------------------------+
                                                ; PCBB+2:            !     ACTUAL PATTERN     !
                                                ;                    +------------------------+
                                                ; PCBB+4:            !   LINK MEMORY ADDRESS  !
                                                ;                    +------------------------+
                                                ; PCBB+6:            !    DATA TRANSFERRED    !
                                                ;   !                !          FROM          !
                                                ;   !                !       LINK MEMORY      !
                                                ;   !                !         BY THE         !
                                                ;   !                !       DMA ENGINE       !
                                                ;   V                !                        !
                                                ; PCBB+4000:         +------------------------+
                                                ;
                                                ;
                                                ;TEST SEQUENCE:
                                                ;       1-LOAD MICROMODULE 'D' IF NOT ALREADY DONE SO
                                                ;       2-LOAD PCSR2 AND PCSR3 WITH ADDRESS OF PCBB
                                                ;       3-CLEAR DMA ENGINE 'TO' BUFFER
                                                ;       4-WAIT FOR MICROMONITOR TO ENTER 'INMON' STATE
                                                ;       5-SELECT MICROTEST #8
                                                ;       6-WAIT FOR 'DNI'
                                                ;       7-CHECK FOR ERROR IN PCSR1
                                                ;       8-VERIFY DATA IN DMA 'TO' BUFFER
                                                ;       9-WRITE '1' TO CLEAR 'DNI'
                                                ;
                                                ;*******************************************************************************

 042470'                                        BGNTST
 042470'                                                                                      T33::
                                                ;
                                                ;CHECK TO SEE IF MODULE 'D' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                                ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                                ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                                ;
 042470'                                          BGNSEG
 042470' 104404                                                                           TRAP    C$BSEG
 042472' 022737  000104  000326'                        CMP     #'D,MICRO             ;HAS MICROCODE MODULE 'D' BEEN LOADED
 042500' 001004                                         BNE     5$                    ;NO
 042502' 122777  000001  135630                         CMPB    #INMON,@PCSR1         ;YES, IS THE MICROMONITOR ACTIVE?
```

```
042510' 001440                          BEQ     20$                             ;YES SKIP LOADING THE MICROMODULE
042512' 012737  000104  000326' 5$:     MOV     #'D,MICRO                       ;GO LOAD MICRO MODULE 'D'
042520' 004737  020340'                 JSR     PC,LODMIC
042524' 103002                          BCC     10$                             ;OK
042526'                                 ESCAPE  TST
042526' 104410                                                                                  TRAP    C$ESCAPE
042530' 000376                                                                                  .WORD   L10153-.
042532' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER                 ;WAIT FOR THE MICROMONITOR
042540' 004737  017316'                 JSR     PC,CHKDNI
042544' 103022                          BCC     20$                             ;OK
042546' 012737  001000' 000310'         MOV     #$DNI,BITNAM
042554' 012737  001277' 000312'         MOV     #$NSET,BITSTA
042562' 012737  001342' 000314'         MOV     #$AFTER,PWHEN
042570' 012737  001357' 000316'         MOV     #$GTCMD,PCOMND
042576'                                 ERRHRD  189,LNKARB,MSG1
042576' 104456                                                                                  TRAP    C$ERHRD
042600' 000275                                                                                  .WORD   189
042602' 003241'                                                                                 .WORD   LNKARB
042604' 012716'                                                                                 .WORD   MSG1
042606'                                 ESCAPE  TST
042606' 104410                                                                                  TRAP    C$ESCAPE
042610' 000316                                                                                  .WORD   L10153-.
042612' 004737  017362'         20$:    JSR     PC,CLRDNI                       ;CLEAR DNI BIT
042616' 103006                          BCC     25$
042620'                                 ERRHRD  190,LNKARB,RACMG7               ;DNI DID NOT CLEAR!
042620' 104456                                                                                  TRAP    C$ERHRD
042622' 000276                                                                                  .WORD   190
042624' 003241'                                                                                 .WORD   LNKARB
042626' 012670'                                                                                 .WORD   RACMG7
042630'                                 ESCAPE TST
042630' 104410                                                                                  TRAP    C$ESCAPE
042632' 000274                                                                                  .WORD   L10153-.
042634'                         25$:
042634'                           ENDSEG
042634'                                                                                 10000$:
042634' 104405                                                                                  TRAP    C$ESEG

                                ;
                                ;SETUP PCBB TO BE JUST AFTER LAST LOCATION USED BY THIS DIAGNOSTIC
                                ;TELL MICROCODE WHERE PCBB IS BY LOADING PCSR2 AND PCSR3 WITH THE PCBB ADDRESS
                                ;
042636' 013777  000324' 135476          MOV     FREMEM,@PCSR2                   ;TELL MICROCODE WHERE PCBB IS
042644' 005077  135474                  CLR     @PCSR3
                                ;
                                ;FIRST CLEAR THE 3 WORDS THAT WILL BE USED BY THE MICROCODE IF AN ERROR OCCURS.
                                ;THEN CLEAR OUT THE BUFFER TO BE USED AS THE DMA ENGINE'S 'TO' BUFFER.
                                ;
042650' 013700  000324'                 MOV     FREMEM,R0                       ;GET A POINTER TO PCBB
042654' 005020                          CLR     (R0)+                           ;HERE IS WHERE MICROCODE WILL PUT
                                                                                ;EXPECTED DATA PATTERN IF ERROR
042656' 005020                          CLR     (R0)+                           ;HERE, THE BAD DATA
042660' 005020                          CLR     (R0)+                           ;AND HERE THE LINK MEMORY ADDRESS

042662' 012701  002000                  MOV     #SIZ1K/2,R1                     ;SIZE OF DMA ENGINE 'TO' BUFFER
042666' 005020                  30$:    CLR     (R0)+                           ;CLEAR THE AREA WHERE THE DMA ENGINE...
042670' 005301                          DEC     R1                              ;WILL LOAD DATA
042672' 001375                          BNE     30$
```

```
042674'                            35$:
042674'                              BGNSEG
042674' 104404                                                                            TRAP    C$BSEG
                                   ;
                                   ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                   ;SELECT MICROTEST #8 BY LOADING PCSR0 COMMAND FIELD BITS WITH A #8.
                                   ;WAIT FOR 'DNI'
                                   ;
042676' 004737  020060'                      JSR     PC,CHKMON            ;WAIT FOR MICROMONITOR
042702' 103006                               BCC     36$                  ;OK
042704'                                      ERRHRD  191,LNKARB,MSG46     ;PRINT ERROR
042704' 104456                                                                            TRAP    C$ERHRD
042706' 000277                                                                            .WORD   191
042710' 003241'                                                                           .WORD   LNKARB
042712' 016666'                                                                           .WORD   MSG46
042714'                                      ESCAPE  TST                  ;LEAVE TEST
042714' 104410                                                                            TRAP    C$ESCAPE
042716' 000210                                                                            .WORD   L10153-.
042720' 012777  000010  135410     36$:      MOV     #8.,@PCSR0           ;TELL T11 TO EXECUTE MICROTEST #8
042726' 012737  000176  000332'              MOV     #2*SECOND,METER      ;WAIT FOR DNI
042734' 004737  017316'                      JSR     PC,CHKDNI
042740' 103021                               BCC     40$
042742' 004737  020132'                      JSR     PC,CHKINT            ;SEE IF ANY ERROR INTERRUPTS OCCURRED
042746' 103006                               BCC     37$                  ;NO, OK
042750'                                      ERRHRD  192,LNKARB,MSG44     ;PRINT ERROR MESSAGE
042750' 104456                                                                            TRAP    C$ERHRD
042752' 000300                                                                            .WORD   192
042754' 003241'                                                                           .WORD   LNKARB
042756' 016442'                                                                           .WORD   MSG44
042760'                                      ESCAPE  TST
042760' 104410                                                                            TRAP    C$ESCAPE
042762' 000144                                                                            .WORD   L10153-.
042764' 012702  000010             37$:      MOV     #8.,R2               ;MICROTEST #8 IS HUNG
042770'                                      ERRHRD  193,LNKARB,MSG12
042770' 104456                                                                            TRAP    C$ERHRD
042772' 000301                                                                            .WORD   193
042774' 003241'                                                                           .WORD   LNKARB
042776' 013466'                                                                           .WORD   MSG12
043000'                                      ESCAPE  TST
043000' 104410                                                                            TRAP    C$ESCAPE
043002' 000124                                                                            .WORD   L10153-.
                                   ;
                                   ;CHECK MICROCODE FINISH STATUS BY CHECKING PCSR1 STATE BITS FOR AN ERROR
                                   ;IF ERROR, PRINT ERROR MESSAGE
                                   ;
043004' 122777  000003  135326     40$:      CMPB    #INERR,@PCSR1        ;DID AND ERROR OCCUR?
043012' 001013                               BNE     50$                  ;NO
043014' 013701  000324'                      MOV     FREMEM,R1            ;YES, GET EXPECTED DATA PATTERN
043020' 013702  000326'                      MOV     FREMEM+2,R2          ;GET BAD DATA
043024' 013703  000330'                      MOV     FREMEM+4,R3          ;GET LINK MEMORY ADDRESS
043030'                                      ERRHRD  194,LNKARB,MSG20     ;PRINT ERROR MESSAGE
043030' 104456                                                                            TRAP    C$ERHRD
043032' 000302                                                                            .WORD   194
043034' 003241'                                                                           .WORD   LNKARB
043036' 014146'                                                                           .WORD   MSG20
043040' 000410                               BR      60$
```

```
                                ;
                                ;CHECK DATA THAT THE DMA ENGINE TRANSFERRED TO UNIBUS MEMORY
                                ;
043042' 013701  000324'         50$:    MOV     FREMEM,R1               ;GET POINTER TO DMA ENGINE BUFFER
043046' 062701  000006                  ADD     #6,R1
043052' 012704  003774                  MOV     #3774,R4                ;NUMBER OF BYTES
043056' 012702  000017                  MOV     #17,R2                  ;DATA THAT SHOULD BE IN BUFFER
043062' 111103                  60$:    MOVB    (R1),R3                 ;GET DATA FROM BUFFER
043064' 120211                          CMPB    R2,(R1)                 ;IS DATA CORRECT?
043066' 001404                          BEQ     70$                     ;YES
043070'                                 ERRHRD  195,LNKARB,MSG21        ;NO, PRINT ERROR
043070' 104456                                                                          TRAP    C$ERHRD
043072' 000303                                                                          .WORD   195
043074' 003241'                                                                         .WORD   LNKARB
043076' 014236'                                                                         .WORD   MSG21
043100' 005201                  70$:    INC     R1
043102' 005304                          DEC     R4
043104' 001366                          BNE     60$                     ;CONTINUE CHECKING
                                                                        ;ALL DONE
                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
                                ;
043106' 004737  017362'         80$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
043112' 103004                          BCC     90$
043114'                                 ERRHRD  196,LNKARB,RACMG7       ;ERROR DNI DID NOT CLEAR!
043114' 104456                                                                          TRAP    C$ERHRD
043116' 000304                                                                          .WORD   196
043120' 003241'                                                                         .WORD   LNKARB
043122' 012670'                                                                         .WORD   RACMG7
043124'                         90$:
043124'                           ENDSEG
043124'                                                                     10001$:
043124' 104405                                                                          TRAP    C$ESEG
043126'                         ENDTST
043126'                                                                     L10153:
043126' 104401                                                                          TRAP    C$ETST
```

```
.SBTTL  TEST 34: STATION ADDRESS PATTERN TEST

;*****************************************************************************************8
;
;WITHOUT EITHER THE PROMISCUIOUS MODE OR THE MULTICAST MODE ENABLED, THE LINK
;LOGIC WILL RECOGNIZE DATAGRAM ADDRESSES ONLY IF THE ADDRESS IS CONTAINED IN
;THE STATION ADDRESS RAM.
;
;WHEN A DATAGRAM ARRIVES, THE LINK LOGIC COMPARES THE DATAGRAM DESTINATION
;ADDRESS FIELD TO THE 12 ADDRESSES WRITTEN IN THE STATION ADDRESS RAM. IF THE
;INCOMING ADDRESS MATCHES ONE OF THESE, THEN THE DATAGRAM WILL BE ACCEPTED BY
;THE LINK. THE 'MATCH' BIT IS SET IN THE TRANSMIT BUFFER AND THE RECEIVING
;PROCESS BEGINS.
;
;THIS TEST WILL VERIFY THAT THE LINK CAN RECOGNIZE A DATAGRAM WHEN THE
;DESTINATION ADDRESS OF THE DATAGRAM MATCHES ONE OF THE ADDRESSES STORED IN
;THE STATION ADDRESS RAM.
;
;THIS TEST WILL USE MICROMODULE 'E' MICROTEST #1.
;PATTERNS WILL BE USED FOR ADDRESSES IN CHECKING THE STATION ADDRESS LOGIC.
;THE PATTERNS WILL BE SUPPLIED TO THE T-11 THROUGH THE PCBB. THE MICROCODE
;WILL BE RESTARTED FOR EACH DIFFERENT PATTERN TO BE TESTED. UPON START-UP, THE
;T-11 PROCESSOR WILL PICK UP THE CURRENT PATTERN/ADDRESS, LOAD THE SAME PATTERN
;INTO ALL 12 LOCATIONS OF THE STATION ADDRESS RAM, FORMAT THE TRANSMIT BUFFER
;AND LOGIC FOR A LOOPBACK, PRESET PCSR1 TO AN ERROR CONDITION, START THE LINK
;AND WAIT FOR THE MATCH BIT IN THE TRANSMIT BUFFER. IF THE MATCH BIT SETS
;THE PCSR1 ERROR CONDITION IS CLEARED AND THE T-11 WAITS FOR BOTH THE
;TRANSMITTER AND RECEIVER INTERRUPTS BEFORE IT SETS 'DNI' TO INDICATE THE
;TEST WAS SUCCESSFUL
;
;THE PCBB WILL BE USED TO PASS THE 48 BIT STATION ADDRESS PATTERN:
;
;                  +--------------------------------+
;PCBB+0:           !       STATION ADDRESS LOW      !
;                  +--------------------------------+
;PCBB+2:           !     STATION ADDRESS MIDDLE     !
;                  +--------------------------------+
;PCBB+4:           !       STATION ADDRESS HIGH     !
;                  +--------------------------------+
;
;THE FOLLOWING PATTERNS WILL BE USED:
;        -ALTERNATING 1'S AND 0'S
;        -ALTERNATING 0'S AND 1'S
;        -PAIR OF 0'S FOLLOWED BY PAIR OF 1'S
;        -FOUR 0'S FOLLOWED BY FOUR 1'S
;        -EIGHT 0'S FOLLOWED BY EIGHT 1'S
;        -SIXTEEN 1'S FOLLOWED BY SIXTEEN 0'S FOLLOWED BY 16 1'S
;        -TWENTYFOUR 1'S FOLLOWED BY TWENTYFOUR 0'S
;
;TEST SEQUENCE:
;        1-LOAD MICRMODULE 'E' IF NOT ALREADY DONE SO
;        2-LOAD PCBB+0,+2,+4 WITH A PATTERN
;        3-WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE
;        4-SELECT MICROTEST #1
;        5-WAIT FOR 'DNI'
;        6-IF NO DNI AND PCSR1 INDICATES ERROR THEN PRINT NO MATCH BIT SET
```

```
                                       ;       7-IF NO DNI AND PCSR1 INDICATES NO ERROR THEN PRINT MATCH BIT SET
                                       ;         BUT DATAGRAM WAS NOT RECEIVED.
                                       ;       8-WRITE '1' TO CLEAR 'DNI'
                                       ;       9-REPEAT STEPS 2-8 FOR ALL SEVEN DATA PATTERNS
                                       ;
                                       ;*************************************************************************************

043130'                                BGNTST
043130'                                                                                    T34::
                                       ;
                                       ;CHECK TO SEE IF MODULE 'E' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                       ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                       ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                       ;
043130'                                  BGNSEG
043130' 104404                                                                              TRAP    C$BSEG
043132' 022737 000105 000326'                  CMP     #'E,MICRO            ;HAS MICROCODE MODULE 'E' BEEN LOADED
043140' 001004                                 BNE     5$                   ;NO
043142' 122777 000001 135170                   CMPB    #INMON,@PCSR1        ;YES, IS THE MICROMONITOR ACTIVE?
043150' 001440                                 BEQ     20$                  ;YES SKIP LOADING THE MICROMODULE
043152' 012737 000105 000326' 5$:              MOV     #'E,MICRO            ;GO LOAD MICRO MODULE 'E'
043160' 004737 020340'                         JSR     PC,LODMIC
043164' 103002                                 BCC     10$                  ;OK
043166'                                        ESCAPE  TST
043166' 104410                                                                              TRAP    C$ESCAPE
043170' 000266                                                                              .WORD   L10154-.
043172' 012737 000176 000332' 10$:             MOV     #2*SECOND,METER      ;WAIT FOR THE MICROMONITOR
043200' 004737 017316'                         JSR     PC,CHKDNI
043204' 103022                                 BCC     20$                  ;OK
043206' 012737 001000' 000310'                 MOV     #$DNI,BITNAM
043214' 012737 001277' 000312'                 MOV     #$NSET,BITSTA
043222' 012737 001342' 000314'                 MOV     #$AFTER,PWHEN
043230' 012737 001357' 000316'                 MOV     #$GTCMD,PCOMND
043236'                                        ERRHRD  197,STAPAT,MSG1
043236' 104456                                                                              TRAP    C$ERHRD
043240' 000305                                                                              .WORD   197
043242' 003305'                                                                             .WORD   STAPAT
043244' 012716'                                                                             .WORD   MSG1
043246'                                        ESCAPE  TST
043246' 104410                                                                              TRAP    C$ESCAPE
043250' 000206                                                                              .WORD   L10154-.
043252' 004737 017362'         20$:            JSR     PC,CLRDNI            ;CLEAR DNI BIT
043256' 103006                                 BCC     25$
043260'                                        ERRHRD  198,STAPAT,RACMG7    ;DNI DID NOT CLEAR!
043260' 104456                                                                              TRAP    C$ERHRD
043262' 000306                                                                              .WORD   198
043264' 003305'                                                                             .WORD   STAPAT
043266' 012670'                                                                             .WORD   RACMG7
043270'                                        ESCAPE TST
043270' 104410                                                                              TRAP    C$ESCAPE
043272' 000164                                                                              .WORD   L10154-.
043274'                                25$:
043274'                                  ENDSEG
043274'                                                                            10000$:
043274' 104405                                                                              TRAP    C$ESEG
                                       ;
```

```
                                        ;SELECT ONE OF THE PATTERNS FROM THE STATION ADDRESS PATTERN TABLE AND LOAD
                                        ;IT INTO THE PCBB
                                        ;
043276' 012701  000534'                         MOV     #SPAT1,R1               ;POINT TO STATION ADDRESS PATTERN TABLE
043302' 012705  000007                          MOV     #7,R5                   ;# OF ADDRESS PATTERNS
043306' 011137  000606'                 27$:    MOV     (R1),PCBB               ;LOAD PCBB WITH AN ADDRESS PATTERN
043312' 016137  000002  000610'                 MOV     2(R1),PCBB+2            ;MIDDLE ORDER
043320' 016137  000004  000612'                 MOV     4(R1),PCBB+4            ;UPPER ORDER
043326'                                   BGNSEG
043326' 104404                                                                                  TRAP    C$BSEG
                                        ;
                                        ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE. SELECT MICROTEST #1
                                        ;BY LOADING PCSR0 COMMAND FIELD BITS WITH A 1. WAIT FOR DNI.
                                        ;
043330' 004737  020060'                         JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
043334' 103006                                  BCC     30$                     ;OK
043336'                                         ERRHRD  199,STAPAT,MSG46        ;PRINT ERROR
043336' 104456                                                                                  TRAP    C$ERHRD
043340' 000307                                                                                  .WORD   199
043342' 003305'                                                                                 .WORD   STAPAT
043344' 016666'                                                                                 .WORD   MSG46
043346'                                         ESCAPE  TST                     ;LEAVE TEST
043346' 104410                                                                                  TRAP    C$ESCAPE
043350' 000106                                                                                  .WORD   L10154-.
043352' 012777  000001  134756  30$:    MOV     #1,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #1
043360' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
043366' 004737  017316'                         JSR     PC,CHKDNI
043372' 103015                                  BCC     40$
                                        ;
                                        ;DNI DID NOT SET WHICH INDICATES THAT THE RECEIVER INTERRUPT NEVER HAPPENED.
                                        ;THE REASON MIGHT BE BECAUSE THE MATCH BIT NEVER SET IN THE TRANSMIT
                                        ;STATUS REGISTER INDICATING THAT THE STATION ADDRESS COMPARATOR FAILED
                                        ;TO RECOGNIZE THE DATAGRAM DESTINATION ADDRESS. THE MICROCODE CHECKS
                                        ;FOR THE MATCH BIT AND IF IT IS NOT SET IT PUTS AN ERROR STATUS IN PCSR1.
                                        ;IF PCSR1 DOES NOT HAVE AN ERROR STATE THEN THE RECEIVER INTERRUPT FAILED
                                        ;FOR SOME OTHER REASON AND ITS NOT THE STATION ADDRESS RECOGNITION'S FAULT.
                                        ;
043374' 004737  020132'                         JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
043400' 103006                                  BCC     35$                     ;NO
043402'                                         ERRHRD  200,STAPAT,MSG44        ;PRINT ERROR MESSAGE
043402' 104456                                                                                  TRAP    C$ERHRD
043404' 000310                                                                                  .WORD   200
043406' 003305'                                                                                 .WORD   STAPAT
043410' 016442'                                                                                 .WORD   MSG44
043412'                                         ESCAPE  TST                     ;LEAVE TST
043412' 104410                                                                                  TRAP    C$ESCAPE
043414' 000042                                                                                  .WORD   L10154-.
043416'                                 35$:    ERRHRD  201,STAPAT,MSG25        ;STATION ADDRESS PATTERN WAS NOT
043416' 104456                                                                                  TRAP    C$ERHRD
043420' 000311                                                                                  .WORD   201
043422' 003305'                                                                                 .WORD   STAPAT
043424' 014614'                                                                                 .WORD   MSG25
                                                                                ;RECOGNIZED! ERROR!
                                        ;
                                        ;WRITE '1' TO CLEAR 'DNI'
                                        ;
```

```
043426' 004737  017362'      40$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
043432' 103004                       BCC     50$
043434'                              ERRHRD  202,STAPAT,RACMG7       ;ERROR DNI DID NOT CLEAR!
043434' 104456                                                                              TRAP    C$ERHRD
043436' 000312                                                                              .WORD   202
043440' 003305'                                                                             .WORD   STAPAT
043442' 012670'                                                                             .WORD   RACMG7
043444'                      50$:
043444'                        ENDSEG
043444'                                                                           10001$:
043444' 104405                                                                              TRAP    C$ESEG

                             ;
                             ;REPEAT THE TEST WITH ALL SEVEN PATTERNS
                             ;
043446' 062701  000006               ADD     #6,R1                   ;POINT TO NEXT PATTERN
043452' 005305                       DEC     R5                      ;TESTED WITH ALL ADDRESS PATTERNS?
043454' 001314                       BNE     27$                     ;NOT YET

043456'                      ENDTST
043456'                                                                           L10154:
043456' 104401                                                                              TRAP    C$ETST
```

```
                                        .SBTTL  TEST 35: STATION ADDRESS REJECTION TEST

                                        ;***************************************************************************
                                        ;
                                        ;THIS TEST WILL VERIFY THAT THE STATION ADDRESS DETECTION LOGIC DOES NOT
                                        ;RECOGNIZE A DATAGRAM WHEN THE DATAGRAM ADDRESS IS NOT CONTAINED IN THE
                                        ;STATION ADDRESS RAM.
                                        ;
                                        ;THE MICROCODE WILL FILL THE STATION ADDRESS RAM WITH 0'S. THE DESTINATION
                                        ;FIELD OF THE TRANSMIT BUFFER IS FILLED WITH 1'S . A TRANSMISSION IS STARTED
                                        ;IN LOOPBACK MODE AND THE T-11 WILL WAIT FOR A RECEIVER INTERRUPT. OF COURSE,
                                        ;THE RECEIVER INTERRUPT SHOULD NEVER HAPPEN BECAUSE THE STATION ADDRESS
                                        ;LOGIC SHOULD NOT GET A SUCCESSFUL COMPARISON BETWEEN 0'S IN THE DESTINATION
                                        ;ADDRESS OF THE INCOMING DATAGRAM AND THE 1'S IN THE STATION ADDRESS RAM.
                                        ;THE T-11 WILL BE PUT INTO A LOOP WAITING FOR A RECEIVER INTERRUPT AND THE
                                        ;DEUNA TIMER IS STARTED. IF THE LOOP IS BROKEN BY THE RECEIVER INTERRUPT
                                        ;AN ERROR WILL BE PRESENTED IN PCSR1 BY THE MICROCODE. IF THE LOOP IS BROKEN
                                        ;BY THE TIMER THEN THE TEST WAS SUCCESSFUL.
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROMODULE 'E' IF NOT ALREADY DONE SO
                                        ;       2-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                        ;       3-SELECT MICROTEST #2
                                        ;       4-WAIT FOR 'DNI'
                                        ;       5-CHECK PCSR1 FOR AN ERROR CONDITION
                                        ;       6-WRITE '1' TO CLEAR 'DNI'
                                        ;
                                        ;***************************************************************************

 043460'                                BGNTST
 043460'                                                                            T35::
                                        ;
                                        ;CHECK TO SEE IF MODULE 'E' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
 043460'                                 BGNSEG
 043460' 104404                                                                           TRAP    C$BSEG
 043462' 022737  000105  000326'            CMP     #'E,MICRO           ;HAS MICROCODE MODULE 'E' BEEN LOADED
 043470' 001004                             BNE     5$                  ;NO
 043472' 122777  000001  134640             CMPB    #INMON,@PCSR1       ;YES, IS THE MICROMONITOR ACTIVE?
 043500' 001440                             BEQ     20$                 ;YES SKIP LOADING THE MICROMODULE
 043502' 012737  000105  000326' 5$:        MOV     #'E,MICRO           ;GO LOAD MICRO MODULE 'E'
 043510' 004737  020340'                    JSR     PC,LODMIC
 043514' 103002                             BCC     10$                 ;OK
 043516'                                    ESCAPE  TST
 043516' 104410                                                                           TRAP    C$ESCAPE
 043520' 000256                                                                           .WORD   L10155-.
 043522' 012737  000176  000332' 10$:       MOV     #2*SECOND,METER     ;WAIT FOR THE MICROMONITOR
 043530' 004737  017316'                    JSR     PC,CHKDNI
 043534' 103022                             BCC     20$                 ;OK
 043536' 012737  001000' 000310'            MOV     #$DNI,BITNAM
 043544' 012737  001277' 000312'            MOV     #$NSET,BITSTA
 043552' 012737  001342' 000314'            MOV     #$AFTER,PWHEN
 043560' 012737  001357' 000316'            MOV     #$GTCMD,PCOMND
 043566'                                    ERRHRD  203,STAREJ,MSG1
```

```
043566' 104456                                                                    TRAP    C$ERHRD
043570' 000313                                                                    .WORD   203
043572' 003351'                                                                   .WORD   STAREJ
043574' 012716'                                                                   .WORD   MSG1
043576'                                   ESCAPE  TST
043576' 104410                                                                    TRAP    C$ESCAPE
043600' 000176                                                                    .WORD   L10155-.
043602' 004737  017362'           20$:    JSR     PC,CLRDNI              ;CLEAR DNI BIT
043606' 103006                            BCC     25$
043610'                                   ERRHRD  204,STAREJ,RACMG7      ;DNI DID NOT CLEAR!
043610' 104456                                                                    TRAP    C$ERHRD
043612' 000314                                                                    .WORD   204
043614' 003351'                                                                   .WORD   STAREJ
043616' 012670'                                                                   .WORD   RACMG7
043620'                                   ESCAPE TST
043620' 104410                                                                    TRAP    C$ESCAPE
043622' 000154                                                                    .WORD   L10155-.
043624'                           25$:
043624'                             ENDSEG
043624'                                                                   10000$:
043624' 104405                                                                    TRAP    C$ESEG

                                  ;
                                  ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE, SELECT MICROTEST #2
                                  ;BY LOADING PCSRO WITH A 2
                                  ;
043626'                             BGNSEG
043626' 104404                                                                    TRAP    C$BSEG
043630' 004737  020060'                   JSR     PC,CHKMON              ;WAIT FOR MICROMONITOR
043634' 103006                            BCC     30$                    ;OK
043636'                                   ERRHRD  205,STAREJ,MSG46       ;PRINT ERROR
043636' 104456                                                                    TRAP    C$ERHRD
043640' 000315                                                                    .WORD   205
043642' 003351'                                                                   .WORD   STAREJ
043644' 016666'                                                                   .WORD   MSG46
043646'                                   ESCAPE  TST                    ;LEAVE TEST
043646' 104410                                                                    TRAP    C$ESCAPE
043650' 000126                                                                    .WORD   L10155-.
043652' 012737  000275  000332'   30$:    MOV     #3*SECOND,METER        ;TIMEOUT MUST BE GREATER THAN TIMER
                                                                         ;PERIOD IN THE MICROCODE.
043660' 012777  000002  134450            MOV     #2,@PCSRO              ;TELL T11 TO EXECUTE MICROTEST #2
043666' 004737  017316'                   JSR     PC,CHKDNI              ;WAIT FOR DNI

043672' 103021                            BCC     40$
043674' 004737  020132'                   JSR     PC,CHKINT              ;SEE IF ANY ERROR INTERRUPTS OCCURRED
043700' 103006                            BCC     35$                    ;NO, OK
043702'                                   ERRHRD  206,STAREJ,MSG44       ;PRINT ERROR MESSAGE
043702' 104456                                                                    TRAP    C$ERHRD
043704' 000316                                                                    .WORD   206
043706' 003351'                                                                   .WORD   STAREJ
043710' 016442'                                                                   .WORD   MSG44
043712'                                   ESCAPE  TST
043712' 104410                                                                    TRAP    C$ESCAPE
043714' 000062                                                                    .WORD   L10155-.
043716' 012702  000002            35$:    MOV     #2,R2                  ;MICROTEST #2 IS HUNG
043722'                                   ERRHRD  207,STAREJ,MSG12
043722' 104456                                                                    TRAP    C$ERHRD
```

```
043724' 000317                                                              .WORD   207
043726' 003351'                                                             .WORD   STAREJ
043730' 013466'                                                             .WORD   MSG12
043732'                          ESCAPE TST
043732' 104410                                                              TRAP    C$ESCAPE
043734' 000042                                                              .WORD   L10155-.
                          ;
                          ;OK 'DNI' SET INDICATING EITHER THE RECEIVER INTERRUPT HAPPENED OR THE TIMER
                          ;BROKE THE LOOP, CHECK PCSR1 TO FIND OUT WHICH
                          ;
043736' 122777 000003 134374  40$:  CMPB    #INERR,@PCSR1          ;DID AN ERROR OCCUR?
043744' 001004                      BNE     50$                    ;NO
043746'                             ERRHRD  208,STAREJ             ;YES, PRINT ERROR MESSAGE
043746' 104456                                                              TRAP    C$ERHRD
043750' 000320                                                              .WORD   208
043752' 003351'                                                             .WORD   STAREJ
043754' 000000                                                              .WORD   0
                          ;
                          ;WRITE '1' TO CLEAR 'DNI'
                          ;
043756' 004737 017362'        50$:  JSR     PC,CLRDNI              ;CLEAR DNI BIT
043762' 103004                      BCC     60$
043764'                             ERRHRD  209,STAREJ,RACMG7      ;ERROR DNI DID NOT CLEAR!
043764' 104456                                                              TRAP    C$ERHRD
043766' 000321                                                              .WORD   209
043770' 003351'                                                             .WORD   STAREJ
043772' 012670'                                                             .WORD   RACMG7
043774'                       60$:
043774'                         ENDSEG
043774'                                                             10001$:
043774' 104405                                                              TRAP    C$ESEG
043776'                       ENDTST
043776'                                                             L10155:
043776' 104401                                                              TRAP    C$ETST
```

```
                                        .SBTTL  TEST 36: STATION ADDRESS RAM POSITION TEST

                                        ;*************************************************************************
                                        ;
                                        ;THE STATION ADDRESS RAM CAN HOLD UP TO 12 STATION ADDRESSES. WHEN A DATAGRAM
                                        ;IS RECEIVED THE STATION ADDRESS COMPARISON LOGIC DOES A BIT-WISE COMPARISON
                                        ;OF ALL 12 RAM STATION ADDRESS WITH THE INCOMING DATAGRAM STATION ADDRESS.
                                        ;
                                        ;THIS TEST WILL VERIFY THAT THE DEUNA CAN RECOGNIZE A STATION ADDRESS
                                        ;REGARDLESS OF THE LOCATION OF THE ADDRESS IN THE STATION ADDRESS RAM.
                                        ;
                                        ;THIS TEST WILL USE MICROMODULE 'E' MICROTEST #3. THE MICROCODE WILL WRITE
                                        ;A STATION ADDRESSES OF ALL 1'S INTO A SINGLE LOCATION OF THE STATION ADDRESS
                                        ;RAM. THE OTHER ELEVEN LOCATION WILL BE LOADED WITH 0'S. A DATAGRAM WITH
                                        ;AN ALL 1'S DESTINATION ADDRESS WILL BE TRANSMITTED IN LOOPBACK MODE. THE TEST
                                        ;WILL VERIFY THAT THE DATAGRAM IS RECEIVED. THE TEST WILL BE REPEATED FOR ALL
                                        ;TWELVE LOCATIONS OF THE STATION ADDRESS RAM.
                                        ;
                                        ;THE MICROTEST WILL BE REPEATED FOR EACH OF THE 12 TEST ITERATIONS. THE PCBB
                                        ;WILL BE USED TO PASS TO THE MICROCODE WHICH POSITION IS TO BE LOADED WITH
                                        ;1'S. WHEN THE STATION ADDRESS IS LOADED, THE STATION ADDRESSES MUST BE
                                        ;ROTATED ORTHOGONALLY, I.E. BIT 0 OF ALL STATION ADDRESSES LOADED TOGETHER,
                                        ;THEN BIT 1, THEN BIT 2 ETC. THIS MAKES IT DIFFICULT TO DESCRIBE THE POSITION
                                        ;OF ANY SINGLE STATION ADDRESS IN TERMS OF AN OFFSET FROM THE RAM BASE ADDRESS.
                                        ;
                                        ;THE PCBB IS FORMATTED AS FOLLOWS:
                                        ;
                                        ;               +-----------------------+
                                        ;PCBB+0:        !  RAM ADDRESS POSTION  !
                                        ;               +-----------------------+
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROMODULE 'E' IF NOT ALREADY DONE SO.
                                        ;       2-LOAD PCBB+0 WITH THE RAM POSTION TO BE TESTED
                                        ;       3-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                        ;       4-SELECT MICROTEST #3
                                        ;       5-WAIT FOR 'DNI'
                                        ;       6-CHECK PCSR1 FOR AN ERROR CONDITION
                                        ;       7-WRITE '1' TO CLEAR 'DNI'
                                        ;       8-REPEAT STEPS 2-7 FOR ALL 12 RAM POSTIONS
                                        ;
                                        ;*************************************************************************

044000'                                 BGNTST
044000'                                                                         T36::
                                        ;
                                        ;CHECK TO SEE IF MODULE 'E' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
044000'                                  BGNSEG
044000' 104404                                                                          TRAP    C$BSEG
044002' 022737  000105  000326'                 CMP     #'E,MICRO               ;HAS MICROCODE MODULE 'E' BEEN LOADED
044010' 001004                                  BNE     5$                      ;NO
044012' 122777  000001  134320                  CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
044020' 001440                                  BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
```

```
044022' 012737  000105  000326' 5$:     MOV     #'E,MICRO               ;GO LOAD MICRO MODULE 'E'
044030' 004737  020340'                 JSR     PC,LODMIC
044034' 103002                          BCC     10$                     ;OK
044036'                                 ESCAPE  TST
044036' 104410                                                                          TRAP    C$ESCAPE
044040' 000306                                                                          .WORD   L10156-.
044042' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
044050' 004737  017316'                 JSR     PC,CHKDNI
044054' 103022                          BCC     20$                     ;OK
044056' 012737  001000' 000310'         MOV     #SDNI,BITNAM
044064' 012737  001277' 000312'         MOV     #SNSET,BITSTA
044072' 012737  001342' 000314'         MOV     #SAFTER,PWHEN
044100' 012737  001357' 000316'         MOV     #SGTCMD,PCOMND
044106'                                 ERRHRD  210,STAPOS,MSG1
044106' 104456                                                                          TRAP    C$ERHRD
044110' 000322                                                                          .WORD   210
044112' 003417'                                                                         .WORD   STAPOS
044114' 012716'                                                                         .WORD   MSG1
044116'                                 ESCAPE  TST
044116' 104410                                                                          TRAP    C$ESCAPE
044120' 000226                                                                          .WORD   L10156-.
044122' 004737  017362'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
044126' 103006                          BCC     25$
044130'                                 ERRHRD  211,STAPOS,RACMG7       ;DNI DID NOT CLEAR!
044130' 104456                                                                          TRAP    C$ERHRD
044132' 000323                                                                          .WORD   211
044134' 003417'                                                                         .WORD   STAPOS
044136' 012670'                                                                         .WORD   RACMG7
044140'                                 ESCAPE TST
044140' 104410                                                                          TRAP    C$ESCAPE
044142' 000204                                                                          .WORD   L10156-.
044144'                         25$:
044144'                           ENDSEG
044144'                                                                         10000$:
044144' 104405                                                                          TRAP    C$ESEG
                                ;
                                ;R1 WILL CONTAIN THE RAM POSTION TO BE TESTED
                                ;
044146' 012701  000001                  MOV     #1,R1                   ;BEGIN WITH RAM POSTION #1
044152'                         30$:
                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE, LOAD PCBB WITH
                                ;THE RAM POSTION TO BE TESTED, SELECT MICROTEST #3 BY LOADING PCSR0 COMMAND
                                ;FIELD WITH A 3, AND WAIT FOR 'DNI' TO SET.
                                ;
044152'                           BGNSEG
044152' 104404                                                                          TRAP    C$BSEG
044154' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
044160' 103006                          BCC     35$                     ;OK
044162'                                 ERRHRD  212,STAPOS,MSG46        ;PRINT ERROR
044162' 104456                                                                          TRAP    C$ERHRD
044164' 000324                                                                          .WORD   212
044166' 003417'                                                                         .WORD   STAPOS
044170' 016666'                                                                         .WORD   MSG46
044172'                                 ESCAPE  TST                     ;LEAVE TEST
044172' 104410                                                                          TRAP    C$ESCAPE
```

```
044174' 000152                                                                  .WORD   L10156-.
044176' 010137  000606'         35$:    MOV     R1,PCBB                 ;TELL MICROCODE WHICH RAM POSTION TO USE
044202' 012777  000003  134126          MOV     #3,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #3
044210' 012737  000275  000332'         MOV     #3*SECOND,METER         ;WAIT FOR DNI
044216' 004737  017316'                 JSR     PC,CHKDNI
044222' 103021                          BCC     40$
044224' 004737  020132'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
044230' 103006                          BCC     36$                     ;NO, OK
044232'                                 ERRHRD  213,STAPOS,MSG44        ;PRINT ERROR MESSAGE
044232' 104456                                                                  TRAP    C$ERHRD
044234' 000325                                                                  .WORD   213
044236' 003417'                                                                 .WORD   STAPOS
044240' 016442'                                                                 .WORD   MSG44
044242'                                 ESCAPE  TST
044242' 104410                                                                  TRAP    C$ESCAPE
044244' 000102                                                                  .WORD   L10156-.
044246' 012702  000003          36$:    MOV     #3,R2                   ;MICROTEST #3 IS HUNG
044252'                                 ERRHRD  214,STAPOS,MSG12
044252' 104456                                                                  TRAP    C$ERHRD
044254' 000326                                                                  .WORD   214
044256' 003417'                                                                 .WORD   STAPOS
044260' 013466'                                                                 .WORD   MSG12
044262'                                 ESCAPE  TST
044262' 104410                                                                  TRAP    C$ESCAPE
044264' 000062                                                                  .WORD   L10156-.
                                ;
                                ;CHECK PCSR1 TO SEE IF THE DATAGRAM WAS RECEIVED
                                ;
044266' 122777  000003  134044  40$:    CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
044274' 001004                          BNE     45$                     ;NO
044276'                                 ERRHRD  215,STAPOS,MSG26        ;PRINT ERROR MESSAGE
044276' 104456                                                                  TRAP    C$ERHRD
044300' 000327                                                                  .WORD   215
044302' 003417'                                                                 .WORD   STAPOS
044304' 015002'                                                                 .WORD   MSG26
044306'                         45$:
044306'                           ENDSEG
044306'                                                                 10001$:
044306' 104405                                                                  TRAP    C$ESEG
                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
                                ;
044310'                           BGNSEG
044310' 104404                                                                  TRAP    C$BSEG
044312' 004737  017362'                 JSR     PC,CLRDNI               ;GO CLEAR DNI
044316' 103006                          BCC     50$                     ;OK
044320'                                 ERRHRD  216,STAPOS,RACMG7       ;ERROR OCCURRED PRINT MESSAGE
044320' 104456                                                                  TRAP    C$ERHRD
044322' 000330                                                                  .WORD   216
044324' 003417'                                                                 .WORD   STAPOS
044326' 012670'                                                                 .WORD   RACMG7
044330'                                 ESCAPE TST
044330' 104410                                                                  TRAP    C$ESCAPE
044332' 000014                                                                  .WORD   L10156-.
044334'                         50$:
044334'                           ENDSEG
```

```
10798  044334'                                                                       10002$:
10799  044334' 104405                                                                        TRAP    C$ESEG
10800                                      ;
10801                                      ;REPEAT TEST FOR ALL 12 RAM POSTIONS
10802                                      ;
10803  044336' 005201                            INC     R1                    ;DO NEXT RAM POSTION
10804  044340' 020127  000014                    CMP     R1,#12.               ;HAVE WE DONE ALL 12 RAM POSTIONS?
10805  044344' 103702                            BLO     30$                   ;NO KEEP GOING
10806  044346'                             ENDTST
10807  044346'                                                                       L10156:
10808  044346' 104401                                                                        TRAP    C$ETST
```

```
                                          .SBTTL  TEST 37: MULTICAST ADDRESS TEST

                                          ;*****************************************************************************
                                          ;
                                          ;MULTICAST ADDRESSING PERMITS THE DEUNA TO RESPOND TO MESSAGES AIMED AT
                                          ;LOGICALLY RELATED DEVICES ON THE NETWORK. THE MSB OF THE DESTINATION ADDRESS
                                          ;OF THESE MESSAGES IS A 1. THIS BIT IS DETECTED BY THE ADDRESS RECOGNITION
                                          ;LOGIC.
                                          ;
                                          ;THIS TEST WILL VERIFY THAT THE DEUNA CAN RECOGNIZE AND ACCEPT MESSAGES WITH
                                          ;THE MULTICAST BIT DESIGNATION.
                                          ;
                                          ;THIS TEST WILL USE MICROMODULE 'E' MICROTEST #4.
                                          ;THE MICROCODE WILL PREPARE A DATAGRAM WITH THE DESTINATION ADDRESS HAVING
                                          ;THE MULTICAST BIT SET. THE DEUNA WILL BE SETUP IN LOOPBACK MODE WITH 'ENABLE
                                          ;ALL MULTICAST'. THE DATAGRAM WILL BE TRANSMITTED AND THE T-11 WILL BE PUT IN
                                          ;A LOOP WAITING FOR A RECEIVER INTERRUPT. THE TIMER WILL INTERRUPT THE LOOP
                                          ;IF THE RECEIVER INTERRUPT DOES NOT OCCUR. IF THIS HAPPENS, PCSR1 WILL INDICATE
                                          ;AN ERROR, OTHERWISE WHEN THE RECEIVER INTERRUPT OCCURS IT WILL BREAK THE LOOP
                                          ;AND PCSR1 WILL INDICATE A SUCCESSFULL COMPLETION OF THE TEST.
                                          ;
                                          ;TEST SEQUENCE:
                                          ;       1-LOAD MICROMODULE 'E' IF NOT ALREADY DONE SO
                                          ;       2-WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE
                                          ;       3-SELECT MICROTEST #4
                                          ;       4-WAIT FOR 'DNI'
                                          ;       5-CHECK PCSR1 FOR AN ERROR CONDITIUN
                                          ;       6-WRITE '1' TO CLEAR 'DNI'
                                          ;
                                          ;*****************************************************************************

044350'                                   BGNTST
044350'                                                                                    T37::
                                          ;
                                          ;CHECK TO SEE IF MODULE 'E' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                          ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                          ;AFTER 'DNI' SETS WRITE '1' TC CLEAR IT.
                                          ;
044350'                                    BGNSEG
044350' 104404                                                                                      TRAP    C$BSEG
044352' 022737  000105  000326'                  CMP     #'E,MICRO              ;HAS MICROCODE MODULE 'E' BEEN LOADED
044360' 001004                                   BNE     5$                     ;NO
044362' 122777  000001  133750                   CMPB    #INMON,@PCSR1          ;YES, IS THE MICROMONITOR ACTIVE?
044370' 001440                                   BEQ     20$                    ;YES SKIP LOADING THE MICROMODULE
044372' 012737  000105  000326'  5$:             MOV     #'E,MICRO              ;GO LOAD MICRO MODULE 'E'
044400' 004737  020340'                          JSR     PC,LODMIC
044404' 103002                                   BCC     10$                    ;OK
044406'                                          ESCAPE  TST
044406' 104410                                                                              TRAP    C$ESCAPE
044410' 000256                                                                              .WORD   L10157-.
044412' 012737  000176  000332'  10$:            MOV     #2*SECOND,METER        ;WAIT FOR THE MICROMONITOR
044420' 004737  017316'                          JSR     PC,CHKDNI
044424' 103022                                   BCC     20$                    ;OK
044426' 012737  001000' 000310'                  MOV     #SDNI,BITNAM
044434' 012737  001277' 000312'                  MOV     #SNSET,BITSTA
044442' 012737  001342' 000314'                  MOV     #SAFTER,PWHEN
```

```
044450' 012737  001357' 000316'         MOV     #$GTCMD,PCOMND
044456'                                 ERRHRD  217,MUCAST,MSG1
044456' 104456                                                                          TRAP    C$ERHRD
044460' 000331                                                                          .WORD   217
044462' 003470'                                                                         .WORD   MUCAST
044464' 012716'                                                                         .WORD   MSG1
044466'                                 ESCAPE  TST
044466' 104410                                                                          TRAP    C$ESCAPE
044470' 000176                                                                          .WORD   L10157-.
044472' 004737  017362'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
044476' 103006                          BCC     25$
044500'                                 ERRHRD  218,MUCAST,RACMG7       ;DNI DID NOT CLEAR!
044500' 104456                                                                          TRAP    C$ERHRD
044502' 000332                                                                          .WORD   218
044504' 003470'                                                                         .WORD   MUCAST
044506' 012670'                                                                         .WORD   RACMG7
044510'                                 ESCAPE TST
044510' 104410                                                                          TRAP    C$ESCAPE
044512' 000154                                                                          .WORD   L10157-.
044514'                         25$:
044514'                           ENDSEG
044514'                                                                         10000$:
044514' 104405                                                                          TRAP    C$ESEG
                                ;
                                ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                ;
044516'                           BGNSEG
044516' 104404                                                                          TRAP    C$BSEG
044520' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
044524' 103006                          BCC     30$                     ;OK
044526'                                 ERRHRD  219,MUCAST,MSG46        ;PRINT ERROR
044526' 104456                                                                          TRAP    C$ERHRD
044530' 000333                                                                          .WORD   219
044532' 003470'                                                                         .WORD   MUCAST
044534' 016666'                                                                         .WORD   MSG46
044536'                                 ESCAPE  TST                     ;LEAVE TEST
044536' 104410                                                                          TRAP    C$ESCAPE
044540' 000126                                                                          .WORD   L10157-.
044542'                         30$:
                                ;
                                ;EXECUTE MICROTEST #4 BY LOADING THE COMMAND FIELD OF PCSR0 WITH A 4
                                ;
044542' 012777  000004  133566          MOV     #4,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #4
                                ;
                                ;WAIT AT LEAST 3 SECONDS FOR TEST TO FINISH BECAUSE MICROCODE WAITS 2 SECONDS
                                ;FOR THE RECEIVER INTERRUPT
                                ;
044550' 012737  000275  000332'         MOV     #3*SECOND,METER         ;WAIT FOR DNI
044556' 004737  017316'                 JSR     PC,CHKDNI
044562' 103021                          BCC     40$
044564' 004737  020132'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
044570' 103006                          BCC     35$                     ;NO, OK
044572'                                 ERRHRD  220,MUCAST,MSG44        ;PRINT ERROR MESSAGE
044572' 104456                                                                          TRAP    C$ERHRD
044574' 000334                                                                          .WORD   220
044576' 003470'                                                                         .WORD   MUCAST
```

```
044600' 016442'                                                                  .WORD   MSG44
044602'                                  ESCAPE  TST
044602' 104410                                                                   TRAP    C$ESCAPE
044604' 000062                                                                   .WORD   L10157-.
044606' 012702  000004          35$:     MOV     #4,R2              ;MICROTEST #4 IS HUNG
044612'                                  ERRHRD  221,MUCAST,MSG12
044612' 104456                                                                   TRAP    C$ERHRD
044614' 000335                                                                   .WORD   221
044616' 003470'                                                                  .WORD   MUCAST
044620' 013466'                                                                  .WORD   MSG12
044622'                                  ESCAPE TST
044622' 104410                                                                   TRAP    C$ESCAPE
044624' 000042                                                                   .WORD   L10157-.
                                ;
                                ;OK, EITHER THE TIMER BROKE THE LOOP OR THE RECEIVER INTERRUPT DID.
                                ;WHICH WAS IT?
                                ;
044626' 122777  000003  133504  40$:     CMPB    #INERR,@PCSR1      ;DID AN ERROR OCCUR?
044634' 001004                           BNE     50$                ;NO
044636'                                  ERRHRD  222,MUCAST         ;YES, PRINT ERROR MESSAGE
044636' 104456                                                                   TRAP    C$ERHRD
044640' 000336                                                                   .WORD   222
044642' 003470'                                                                  .WORD   MUCAST
044644' 000000                                                                   .WORD   0
                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
                                ;
044646' 004737  017362'         50$:     JSR     PC,CLRDNI          ;CLEAR DNI BIT
044652' 103004                           BCC     60$
044654'                                  ERRHRD  223,MUCAST,RACMG7  ;ERROR DNI DID NOT CLEAR!
044654' 104456                                                                   TRAP    C$ERHRD
044656' 000337                                                                   .WORD   223
044660' 003470'                                                                  .WORD   MUCAST
044662' 012670'                                                                  .WORD   RACMG7
044664'                         60$:
044664'                           ENDSEG
044664'                                                                  10001$:
044664' 104405                                                                   TRAP    C$ESEG
044666'                         ENDTST
044666'                                                                  L10157:
044666' 104401                                                                   TRAP    C$ETST
```

```
                        .SBTTL  TEST 38: CRC DATA PATTERN TEST

                        ;*************************************************************************8
                        ;
                        ;THE LINK MODULE HAS HARDWARE TO GUARANTEE THAT DATAGRAMS HAVE NOT BEEN
                        ;CORRUPTED DURING TRANSMISSION AND RECEPTION. THE HARDWARE GENERATES A CRC
                        ;FOR DATAGRAMS TRANSMITTED AND VERIFIES THE CRC FOR DATAGRAMS RECEIVED.
                        ;THE CRC IS A 32 BIT NUMBER GENERATED BY DIVIDING THE DATAGRAM BIT STREAM BY A
                        ;CRC POLYNOMIAL. THE DIVISION RESULTS IN A UNIQUE NUMBER THAT CAN ONLY BE
                        ;REPRODUCED IN CRC CALCULATIONS IF THE BIT STREAM EXACTLY MATCHES THE ORIGINAL.
                        ;THE CRC IS CALCULATED DURING DATAGRAM TRANSMISSION AND IS APPENDED TO THE
                        ;PACKET. THE CRC IS TRANSMITTED AS PART OF THE PACKET. THE CRC IS AGAIN
                        ;CALCULATED WHEN THE DATAGRAM IS RECEIVED AND THE CALCULATED IS COMPARED TO
                        ;THE CRC TRANSMITTED. IF THE DATAGRAM HAS BEEN FAITHFULLY TRANSMITTED, THE
                        ;CRC'S SHOULD MATCH EXACTLY.
                        ;
                        ;THE DEUNA CALCULATES THE CRC WITH DEDICATED CRC LOGIC. THE LOGIC IS EITHER
                        ;DEDICATED TO THE CALCULATION OF THE OUTGOING DATAGRAM OR THE CALCULATION OF
                        ;THE INCOMING DATAGRAM, BUT NOT BOTH.
                        ;
                        ;THIS TEST WILL VERIFY THE OPERATION OF THE CRC CALCULATION CIRCUITRY.
                        ;MICROMODULE 'F' MICROTEST #1 WILL BE USED.
                        ;THE MICROCODE WILL TRANSMIT DATAGRAMS IN LOOPBACK MODE. THE CRC HARDWARE WILL
                        ;BE DEDICATED TO THE TRANSMITTER. WHEN THE DATAGRAM IS RECEIVED THE T-11 WILL
                        ;CALCULATE A CRC ON THE DATA RECEIVED (INCLUDING THE TRANSMITTED CRC).
                        ;THE RESULT OF THIS CALCULATION WILL BE A 32 BIT CONSTANT. THIS CONSTANT IS
                        ;THEN COMPARED TO WHAT WAS EXPECTED AND IF THEY DO NOT MATCH. AN ERROR IS
                        ;PLACED IN PCSR1.
                        ;
                        ;PATTERNS WILL BE PASSED TO THE MICROCODE THROUGH THE PCBB. THE MICROCODE WILL
                        ;FILL THE TRANSMIT BUFFER WITH THIS PATTERN BEFORE EACH TRANSMISSION TAKES
                        ;PLACE.
                        ;
                        ;THE PCBB WILL BE FORMATTED AS FOLLOWS:
                        ;
                        ;               +--------------+
                        ;PCBB+0:        ! DATA PATTERN !
                        ;               +--------------+
                        ;
                        ;TEST SEQUENCE:
                        ;       1-LOAD MICROMODULE 'F' IF NOT ALREADY DONE SO
                        ;       2-PLACE A DATA PATTERN IN PCBB+0
                        ;       3-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                        ;       4-SELECT MICROTEST #1
                        ;       5-WAIT FOR 'DNI'
                        ;       6-CHECK PCSR1 FOR AN ERROR CONDITION
                        ;       7-WRITE '1' TO CLEAR 'DNI'
                        ;
                        ;*************************************************************************8

        044670'         BGNTST
        044670'                                                                 T38::
                        ;
                        ;CHECK TO SEE IF MODULE 'F' HAS BEEN LOADED. IF NOT LOAD IT INTO
                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
```

```
                                        ;
044670'                                   BGNSEG
044670' 104404                                                                                          TRAP    C$BSEG
044672' 022737  000106  000326'                 CMP     #'F,MICRO                   ;HAS MICROCODE MODULE 'F' BEEN LOADED
044700' 001004                                  BNE     5$                          ;NO
044702' 122777  000001  133430                  CMPB    #INMON,@PCSR1               ;YES, IS THE MICROMONITOR ACTIVE?
044710' 001440                                  BEQ     20$                         ;YES SKIP LOADING THE MICROMODULE
044712' 012737  000106  000326' 5$:             MOV     #'F,MICRO                   ;GO LOAD MICRO MODULE 'F'
044720' 004737  020340'                         JSR     PC,LODMIC
044724' 103002                                  BCC     10$                         ;OK
044726'                                         ESCAPE  TST
044726' 104410                                                                                          TRAP    C$ESCAPE
044730' 000276                                                                                          .WORD   L10160-.
044732' 012737  000176  000332' 10$:            MOV     #2*SECOND,METER             ;WAIT FOR THE MICROMONITOR
044740' 004737  017316'                         JSR     PC,CHKDNI
044744' 103022                                  BCC     20$                         ;OK
044746' 012737  001000' 000310'                 MOV     #$DNI,BITNAM
044754' 012737  001277' 000312'                 MOV     #$NSET,BITSTA
044762' 012737  001342' 000314'                 MOV     #$AFTER,PWHEN
044770' 012737  001357' 000316'                 MOV     #$GTCMD,PCOMND
044776'                                         ERRHRD  224,CRCDAT,MSG1
044776' 104456                                                                                          TRAP    C$ERHRD
045000' 000340                                                                                          .WORD   224
045002' 003526'                                                                                         .WORD   CRCDAT
045004' 012716'                                                                                         .WORD   MSG1
045006'                                         ESCAPE  TST
045006' 104410                                                                                          TRAP    C$ESCAPE
045010' 000216                                                                                          .WORD   L10160-.
045012' 004737  017362'         20$:            JSR     PC,CLRDNI                   ;CLEAR DNI BIT
045016' 103006                                  BCC     25$
045020'                                         ERRHRD  225,CRCDAT,RACMG7           ;DNI DID NOT CLEAR!
045020' 104456                                                                                          TRAP    C$ERHRD
045022' 000341                                                                                          .WORD   225
045024' 003526'                                                                                         .WORD   CRCDAT
045026' 012670'                                                                                         .WORD   RACMG7
045030'                                         ESCAPE TST
045030' 104410                                                                                          TRAP    C$ESCAPE
045032' 000174                                                                                          .WORD   L10160-.
045034'                         25$:
045034'                                   ENDSEG
045034'                                                                                 10000$:
045034' 104405                                                                                          TRAP    C$ESEG
                                        ;
                                        ;SELECT A DATA PATTERN FROM A TABLE OF PREDEFINED DATA PATTERNS
                                        ;AND LOAD IT INTO PCBB+0 FOR THE MICROCODE
                                        ;
045036' 012701  000520'                         MOV     #PATERN,R1                  ;GET ADDRESS OF DATA PATTERN TABLE
045042' 012705  000005                          MOV     #5,R5                       ;# OF DATA PATTERNS
045046' 012137  000606'         27$:            MOV     (R1)+,PCBB                  ;LOAD PCBB WITH A DATA PATTERN
045052'                                   BGNSEG
045052' 104404                                                                                          TRAP    C$BSEG
                                        ;
                                        ;WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                        ;
045054' 004737  020060'                         JSR     PC,CHKMON                   ;WAIT FOR MICROMONITOR
045060' 103006                                  BCC     30$                         ;OK
```

```
045062'                                     ERRHRD  227,CRCDAT,MSG46        ;PRINT ERROR
045062' 104456                                                                                   TRAP    C$ERHRD
045064' 000343                                                                                   .WORD   227
045066' 003526'                                                                                  .WORD   CRCDAT
045070' 016666'                                                                                  .WORD   MSG46
045072'                                     ESCAPE  TST                     ;LEAVE TEST
045072' 104410                                                                                   TRAP    C$ESCAPE
045074' 000132                                                                                   .WORD   L10160-.
045076'                             30$:
                                    ;
                                    ;EXECUTE MICROTEST #1 BY LOADING PCSR0 COMMAND FIELD WITH A 1
                                    ;WAIT FOR DNI
                                    ;
045076' 012777  000001  133232              MOV     #1,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #1
045104' 012737  000176  000332'             MOV     #2*SECOND,METER         ;WAIT FOR DNI
045112' 004737  017316'                     JSR     PC,CHKDNI
045116' 103021                              BCC     40$
045120' 004737  020132'                     JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
045124' 103006                              BCC     35$                     ;NO, OK
045126'                                     ERRHRD  228,CRCDAT,MSG44        ;PRINT ERROR MESSAGE
045126' 104456                                                                                   TRAP    C$ERHRD
045130' 000344                                                                                   .WORD   228
045132' 003526'                                                                                  .WORD   CRCDAT
045134' 016442'                                                                                  .WORD   MSG44
045136'                                     ESCAPE  TST
045136' 104410                                                                                   TRAP    C$ESCAPE
045140' 000066                                                                                   .WORD   L10160-.
045142' 012702  000005          35$:        MOV     #5,R2                   ;MICROTEST #1 IS HUNG
045146'                                     ERRHRD  229,CRCDAT,MSG12
045146' 104456                                                                                   TRAP    C$ERHRD
045150' 000345                                                                                   .WORD   229
045152' 003526'                                                                                  .WORD   CRCDAT
045154' 013466'                                                                                  .WORD   MSG12
045156'                                     ESCAPE  TST
045156' 104410                                                                                   TRAP    C$ESCAPE
045160' 000046                                                                                   .WORD   L10160-.
                                    ;
                                    ;CHECK PCSR1 TO SEE IF THE MICROCODE DETECTED AN ERROR IN THE CRC CALCULATION
                                    ;
045162' 122777  000003  133150  40$:        CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
045170' 001004                              BNE     50$                     ;NO
045172'                                     ERRHRD  230,CRCDAT              ;PRINT ERROR MESSAGE
045172' 104456                                                                                   TRAP    C$ERHRD
045174' 000346                                                                                   .WORD   230
045176' 003526'                                                                                  .WORD   CRCDAT
045200' 000000                                                                                   .WORD   0
                                    ;
                                    ;WRITE '1' TO CLEAR 'DNI'
                                    ;
045202' 004737  017362'         50$:        JSR     PC,CLRDNI               ;CLEAR DNI BIT
045206' 103004                              BCC     55$
045210'                                     ERRHRD  231,CRCDAT,RACMG7       ;ERROR DNI DID NOT CLEAR!
045210' 104456                                                                                   TRAP    C$ERHRD
045212' 000347                                                                                   .WORD   231
045214' 003526'                                                                                  .WORD   CRCDAT
045216' 012670'                                                                                  .WORD   RACMG7
```

```

045220'                    55$:
045220'                      ENDSEG
045220'                                                                                   10001$:
045220' 104405                                                                                    TRAP    C$ESEG
                           ;
                           ;REPEAT THE TEST EACH TIME WITH A NEW DATA PATTERN
                           ;
045222' 005305                     DEC     R5                         ;HAVE WE TESTED WITH ALL DATA PATTERNS?
045224' 001310                     BNE     27$                        ;NOT YET
045226'                    ENDTST
045226'                                                                                   L10160:
045226' 104401                                                                                    TRAP    C$ETST
```

```
                                          .SBTTL  TEST 39: CRC ERROR TEST
                                          ;*********************************************************************
                                          ;
                                          ;THIS TEST WILL VERIFY THAT THE LINK CRC CIRCUITRY CAN DETECT A BAD CRC.
                                          ;
                                          ;MICROMODULE 'F' MICROTEST #2 WILL BE USED. THE MICROCODE WILL TRANSMIT
                                          ;DATAGRAMS IN LOOPBACK MODE. EACH DATAGRAM WILL HAVE AN ERRONEOUS CRC
                                          ;APPENDED TO THE DATA FIELD. THE DEUNA CRC LOGIC WILL BE SETUP SUCH THAT
                                          ;THE CRC LOGIC WILL BE DEDICATED TO THE RECEIVER. THIS IS EXPECTED TO CAUSE
                                          ;A CRC ERROR.
                                          ;
                                          ;THE DATA FIELDS OF EACH DATAGRAM WILL CONSIST OF PATTERNS. THE PATTERNS
                                          ;WILL BE PASSED TO THE MICROCODE VIA THE PCBB.
                                          ;
                                          ;AFTER THE RECEIVER INTERRUPT THE MICROCODE WILL PASS THE RECEIVER STATUS WORD
                                          ;0 BACK VIA PCBB+2. THE CRC BIT IN THIS WORD IS CHECKED TO SEE IF IT IS SET.
                                          ;
                                          ;THE PCBB IS FORMATTED AS FOLLOWS:
                                          ;
                                          ;              +-----------------------+
                                          ;PCBB+0:       !     DATA PATTERN      !
                                          ;              +-----------------------+
                                          ;PCBB+2:       ! RECEIVER STATUS WORD  !
                                          ;              +-----------------------+
                                          ;
                                          ;TEST SEQUENCE:
                                          ;       1-LOAD MICROMODULE 'F' IF NOT ALREADY DONE SO
                                          ;       2-PLACE A DATA PATTERN IN PCBB+0
                                          ;       3-WAIT FOR THE MICROCODE TO ENTER THE 'INMON' STATE
                                          ;       4-SELECT MICROTEST #2
                                          ;       5-WAIT FOR 'DNI'
                                          ;       6-VERIFY CRC BIT AND ERROR SUMMARY BIT SET IN PCBB+2
                                          ;       7-WRITE '1' TO CLEAR 'DNI'
                                          ;
                                          ;*********************************************************************
        045230'                           BGNTST
        045230'                                                                      T39::
                                          ;
                                          ;CHECK TO SEE IF MODULE 'F' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                          ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                          ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                          ;
        045230'                            BGNSEG
        045230' 104404                                                                       TRAP    C$BSEG
        045232' 022737  000106  000326'         CMP     #'F,MICRO               ;HAS MICROCODE MODULE 'F' BEEN LOADED
        045240' 001004                          BNE     5$                      ;NO
        045242' 122777  000001  133070          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
        045250' 001440                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
        045252' 012737  000106  000326' 5$:     MOV     #'F,MICRO               ;GO LOAD MICRO MODULE 'F'
        045260' 004737  020340'                 JSR     PC,LODMIC
        045264' 103002                          BCC     10$                     ;OK
        045266'                                 ESCAPE  TST
        045266' 104410                                                                       TRAP    C$ESCAPE
        045270' 000326                                                                       .WORD   L10161-.
```

```
045272' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
045300' 004737  017316'                 JSR     PC,CHKDNI
045304' 103022                          BCC     20$                     ;OK
045306' 012737  001000' 000310'         MOV     #SDNI,BITNAM
045314' 012737  001277' 000312'         MOV     #SNSET,BITSTA
045322' 012737  001342' 000314'         MOV     #SAFTER,PWHEN
045330' 012737  001357' 000316'         MOV     #SGTCMD,PCOMND
045336'                                 ERRHRD  232,CRCERR,MSG1
045336' 104456                                                                          TRAP    C$ERHRD
045340' 000350                                                                          .WORD   232
045342' 003563'                                                                         .WORD   CRCERR
045344' 012716'                                                                         .WORD   MSG1
045346'                                 ESCAPE  TST
045346' 104410                                                                          TRAP    C$ESCAPE
045350' 000246                                                                          .WORD   L10161-.
045352' 004737  017362'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
045356' 103006                          BCC     25$
045360'                                 ERRHRD  233,CRCERR,RACMG7       ;DNI DID NOT CLEAR!
045360' 104456                                                                          TRAP    C$ERHRD
045362' 000351                                                                          .WORD   233
045364' 003563'                                                                         .WORD   CRCERR
045366' 012670'                                                                         .WORD   RACMG7
045370'                                 ESCAPE TST
045370' 104410                                                                          TRAP    C$ESCAPE
045372' 000224                                                                          .WORD   L10161-.
045374'                         25$:
045374'                           ENDSEG
045374'                                                                         10000$:
045374' 104405                                                                          TRAP    C$ESEG
                                ;
                                ;CLEAR PCBB+2, GET A DATA PATTERN FROM A LIST OF PRESELECTED DATA PATTERNS
                                ;AND PLACE IT IN PCBB+0 FOR THE MICROCODE
                                ;
045376' 005037  000610'                 CLR     PCBB+2                  ;HERE IS WHERE MICROCODE WILL PUT...
                                                                        ;STATUS WORD
045402' 012701  000520'                 MOV     #PATERN,R1              ;GET ADDRESS OF DATA PATTERN TABLE
045406' 012705  000005                  MOV     #5,R5                   ;# OF DATA PATTERNS
045412' 012137  000606'         27$:    MOV     (R1)+,PCBB              ;LOAD PCBB WITH A DATA PATTERN
                                ;
                                ;WAIT FOR THE MICROMONITOR TO BECOME READY
                                ;
045416'                           BGNSEG
045416' 104404                                                                          TRAP    C$BSEG
045420' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
045424' 103006                          BCC     30$                     ;OK
045426'                                 ERRHRD  234,CRCERR,MSG46        ;PRINT ERROR
045426' 104456                                                                          TRAP    C$ERHRD
045430' 000352                                                                          .WORD   234
045432' 003563'                                                                         .WORD   CRCERR
045434' 016666'                                                                         .WORD   MSG46
045436'                                 ESCAPE  TST                     ;LEAVE TEST
045436' 104410                                                                          TRAP    C$ESCAPE
045440' 000156                                                                          .WORD   L10161-.
045442'                         30$:
                                ;
                                ;EXECUTE MICROTEST #2 BY LOADING THE COMMAND FIELD OF PCSRO WITH A 2
```

```
                                        ;WAIT FOR 'DNI'
                                        ;
045442' 012777  000002  132666                  MOV     #2,@PCSRO               ;TELL T11 TO EXECUTE MICROTEST #2
045450' 012737  000176  000332'                 MOV     #2*SECOND,METER         ;WAIT FOR DNI
045456' 004737  017316'                         JSR     PC,CHKDNI
045462' 103021                                  BCC     40$
045464' 004737  020132'                         JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
045470' 103006                                  BCC     35$                     ;NO, OK
045472'                                         ERRHRD  235,CRCERR,MSG44        ;PRINT ERROR MESSAGE
045472' 104456                                                                                          TRAP    C$ERHRD
045474' 000353                                                                                          .WORD   235
045476' 003563'                                                                                         .WORD   CRCERR
045500' 016442'                                                                                         .WORD   MSG44
045502'                                         ESCAPE  TST
045502' 104410                                                                                          TRAP    C$ESCAPE
045504' 000112                                                                                          .WORD   L10161-.
045506' 012702  000002                  35$:    MOV     #2,R2                   ;MICROTEST #2 IS HUNG
045512'                                         ERRHRD  236,CRCERR,MSG12
045512' 104456                                                                                          TRAP    C$ERHRD
045514' 000354                                                                                          .WORD   236
045516' 003563'                                                                                         .WORD   CRCERR
045520' 013466'                                                                                         .WORD   MSG12
045522'                                         ESCAPE  TST
045522' 104410                                                                                          TRAP    C$ESCAPE
045524' 000072                                                                                          .WORD   L10161-.
                                        ;
                                        ;OK, TEST IS COMPLETED, NOW CHECK PCBB+2. PCBB+2 CONTAINS THE RECEIVER STATUS
                                        ;WORD, IT SHOULD HAVE THE CRC ERROR BIT SET AND THE ERROR SUMMARY BIT SET
                                        ;
045526' 013704  000610'                 40$:    MOV     PCBB+2,R4               ;THIS IS THE RECEIVER STATUS WORD
045532' 012703  004000                          MOV     #BIT11,R3               ;BIT CRC SHOULD BE SET
045536' 030304                                  BIT     R3,R4                   ;IS CRC BIT SET?
045540' 001004                                  BNE     45$                     ;YES, OK
045542'                                         ERRHRD  237,CRCERR,MSG27        ;NO, PRINT ERROR MESSAGE
045542' 104456                                                                                          TRAP    C$ERHRD
045544' 000355                                                                                          .WORD   237
045546' 003563'                                                                                         .WORD   CRCERR
045550' 015046'                                                                                         .WORD   MSG27
045552' 012703  040000                  45$:    MOV     #BIT14,R3               ;ERROR SUMMARY BIT SHOULD BE SET
045556' 030304                                  BIT     R3,R4                   ;IS ERROR SUMMARY BIT SET?
045560' 001004                                  BNE     50$                     ;YES, GOOD
045562'                                         ERRHRD  238,CRCERR,MSG28        ;NO, PRINT ERROR MESSAGE
045562' 104456                                                                                          TRAP    C$ERHRD
045564' 000356                                                                                          .WORD   238
045566' 003563'                                                                                         .WORD   CRCERR
045570' 015114'                                                                                         .WORD   MSG28
                                        ;
                                        ;WRITE '1' TO CLEAR 'DNI'
                                        ;
045572' 004737  017362'                 50$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
045576' 103004                                  BCC     55$
045600'                                         ERRHRD  239,CRCERR,RACMG7       ;ERROR DNI DID NOT CLEAR!
045600' 104456                                                                                          TRAP    C$ERHRD
045602' 000357                                                                                          .WORD   239
045604' 003563'                                                                                         .WORD   CRCERR
045606' 012670'                                                                                         .WORD   RACMG7
```

```
045610'                     55$:
045610'                       ENDSEG
045610'                                                                    10001$:
045610' 104405                                                                     TRAP    C$ESEG
                            ;
                            ;REPEAT TEST WITH ALL DATA PATTERNS
                            ;
045612' 005305                      DEC     R5                      ;HAVE WE TESTED WITH ALL DATA PATTERNS?
045614' 001276                      BNE     27$                     ;NOT YET

045616'                     ENDTST
045616'                                                                    L10161:
045616' 104401                                                                     TRAP    C$ETST
```

```
                                        .SBTTL  TEST 40: CRC PATTERN LENGTH TEST

                                        ;*******************************************************************************
                                        ;
                                        ;THIS TEST WILL VERIFY THAT THE RECEIVE CRC HARDWARE CAN CALCULATE CRC FOR
                                        ;DATAGRAMS OF VARYING LENGTHS.
                                        ;
                                        ;DATAGRAMS WILL BE TRANSMITTED FOR THE TRANSMIT BUFFER TO THE RECEIVE BUFFER
                                        ;IN LOOPBACK MODE. THE TRANSMIT CRC WILL BE DISABLED WHICH WILL ASSIGN THE
                                        ;CRC LOGIC TO CALCULATION OF INCOMING DATAGRAMS. THE CRC FOR TRANSMIT DATAGRAMS
                                        ;WILL BE CALCULATED BY THE MICROCODE. IT IS EXPECTED THAT THE CRC LOGIC WILL
                                        ;VERIFY THE CRC APPENDED TO THE DATAGRAM AS IT IS BEING RECEIVED.
                                        ;
                                        ;PATTERNS WILL BE USED TO FILL THE DATAGRAM DATA FIELD. THE PATTERNS WILL BE
                                        ;PASSED TO THE MICROCODE THROUGH THE PCBB ALONG WITH THE BYTE COUNT TO BE USED.
                                        ;
                                        ;AFTER THE RECEPTION OF THE DATAGRAM THE RECEIVER STATUS WORD WILL BE PASSED
                                        ;BACK VIA THE PCBB SO IT CAN BE CHECKED
                                        ;
                                        ;THE PCBB IS FORMATTED AS FOLLOWS:
                                        ;
                                        ;               +---------------------+
                                        ;PCBB+0:        !    DATA PATTERN     !
                                        ;               +---------------------+
                                        ;PCBB+2:        !     BYTE COUNT      !
                                        ;               +---------------------+
                                        ;PCBB+4:        ! RECEIVE STATUS WORD !
                                        ;               +---------------------+
                                        ;
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROMODULE 'F' IF NOT ALREADY DONE SO
                                        ;       2-PLACE A PATTERN IN PCBB+0
                                        ;       3-PREPARE A MINIMUM BYTE COUNT
                                        ;       4-PLACE BYTE COUNT IN PCBB+4
                                        ;       5-WAIT FOR MICROMONITOR TO ENTER 'INMON' STATE
                                        ;       6-SELECT MICROTEST #3
                                        ;       7-WAIT FOR 'DNI'
                                        ;       8-VERIFY NO ERRORS IN PCBB+4
                                        ;       9-WRITE '1' TO CLEAR 'DNI'
                                        ;       10-MULTIPLY BYTE COUNT BY 2
                                        ;       11-REPEAT STEPS 4-10 UNTIL MAXIMUM BYTE COUNT IS REACHED
                                        ;       12-REPEAT STEPS 2-11 WITH ALL DATA PATTERNS
                                        ;
                                        ;*******************************************************************************

 045620'                                BGNTST
 045620'                                                                        T40::
                                        ;
                                        ;CHECK TO SEE IF MODULE 'F' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
 045620'                                 BGNSEG
 045620' 104404                                                                         TRAP    C$BSEG
 045622' 022737  000106  000326'           CMP     #'F,MICRO            ;HAS MICROCODE MODULE 'F' BEEN LOADED
 045630' 001004                            BNE     5$                   ;NO
```

```
045632' 122777  000001  132500          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
045640' 001440                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
045642' 012737  000106  000326' 5$:     MOV     #'F,MICRO               ;GO LOAD MICRO MODULE 'F'
045650' 004737  020340'                 JSR     PC,LODMIC
045654' 103002                          BCC     10$                     ;OK
045656'                                 ESCAPE  TST
045656' 104410                                                                  TRAP    C$ESCAPE
045660' 000320                                                                  .WORD   L10162-.
045662' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
045670' 004737  017316'                 JSR     PC,CHKDNI
045674' 103022                          BCC     20$                     ;OK
045676' 012737  001000' 000310'         MOV     #$DNI,BITNAM
045704' 012737  001277' 000312'         MOV     #$NSET,BITSTA
045712' 012737  001342' 000314'         MOV     #$AFTER,PWHEN
045720' 012737  001357' 000316'         MOV     #$GTCMD,PCOMND
045726'                                 ERRHRD  240,CRCPAT,MSG1
045726' 104456                                                                  TRAP    C$ERHRD
045730' 000360                                                                  .WORD   240
045732' 003611'                                                                 .WORD   CRCPAT
045734' 012716'                                                                 .WORD   MSG1
045736'                                 ESCAPE  TST
045736' 104410                                                                  TRAP    C$ESCAPE
045740' 000240                                                                  .WORD   L10162-.
045742' 004737  017362'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
045746' 103006                          BCC     25$
045750'                                 ERRHRD  241,CRCPAT,RACMG7       ;DNI DID NOT CLEAR!
045750' 104456                                                                  TRAP    C$ERHRD
045752' 000361                                                                  .WORD   241
045754' 003611'                                                                 .WORD   CRCPAT
045756' 012670'                                                                 .WORD   RACMG7
045760'                                 ESCAPE  TST
045760' 104410                                                                  TRAP    C$ESCAPE
045762' 000216                                                                  .WORD   L10162-.
045764'                         25$:
045764'                           ENDSEG
045764'                                                             10000$:
045764' 104405                                                                  TRAP    C$ESEG
                                ;
                                ;POINT TO LIST OF DATA PATTERNS
                                ;
045766' 012701  000520'                 MOV     #PATERN,R1              ;GET ADDRESS OF DATA PATTERN TABLE
045772' 012705  000005                  MOV     #5,R5                   ;# OF DATA PATTERNS
                                ;
                                ;GET A DATA PATTERN FROM THE LIST AND PLACE IT IN PCBB+0
                                ;
045776'                         27$:
045776' 012137  000606'                 MOV     (R1)+,PCBB              ;LOAD PCBB WITH A DATA PATTERN
046002'                           BGNSEG
046002' 104404                                                                  TRAP    C$BSEG
                                ;
                                ;GENERATE A MINIMUM BYTE COUNT
                                ;
046004' 012703  000100                  MOV     #BIT6,R3                ;R3 WILL BE BYTE COUNT
046010'                         28$:
                                ;
                                ;LOAD BYTE COUNT INTO THE PCBB+4
```

```
                                                ;
        046010'                                  BGNSEG
        046010' 104404                                                                          TRAP    C$BSEG
        046012' 010337  000610'                          MOV     R3.PCBB+2               ;LOAD BYTE COUNT INTO PCBB
                                                ;
                                                ;WAIT FOR THE MICROCODE TO ENTER THE 'INMON' STATE
                                                ;EXECUTE MICROTEST #3 BY LOADING THE COMMAND FIELD OF PCSRO WITH A 3
                                                ;WAIT FOR 'DNI'
                                                ;
        046016' 004737  020060'                          JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
        046022' 103006                                   BCC     30$                     ;OK
        046024'                                          ERRHRD  242,CRCPAT,MSG46        ;PRINT ERROR
        046024' 104456                                                                          TRAP    C$ERHRD
        046026' 000362                                                                          .WORD   242
        046030' 003611'                                                                         .WORD   CRCPAT
        046032' 016666'                                                                         .WORD   MSG46
        046034'                                          ESCAPE  TST                     ;LEAVE TEST
        046034' 104410                                                                          TRAP    C$ESCAPE
        046036' 000142                                                                          .WORD   L10162-.
        046040' 012777  000003  132270  30$:     MOV     #3,@PCSRO               ;TELL T11 TO EXECUTE MICROTEST #3
        046046' 012737  000176  000332'          MOV     #2*SECOND,METER         ;WAIT FOR DNI
        046054' 004737  017316'                  JSR     PC,CHKDNI
        046060' 103021                           BCC     40$
        046062' 004737  020132'                  JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
        046066' 103006                           BCC     35$                     ;NO, OK
        046070'                                  ERRHRD  243,CRCPAT,MSG44        ;PRINT ERROR MESSAGE
        046070' 104456                                                                          TRAP    C$ERHRD
        046072' 000363                                                                          .WORD   243
        046074' 003611'                                                                         .WORD   CRCPAT
        046076' 016442'                                                                         .WORD   MSG44
        046100'                                  ESCAPE  TST
        046100' 104410                                                                          TRAP    C$ESCAPE
        046102' 000076                                                                          .WORD   L10162-.
        046104' 012702  000003          35$:     MOV     #3,R2                   ;MICROTEST #3 IS HUNG
        046110'                                  ERRHRD  244,CRCPAT,MSG12
        046110' 104456                                                                          TRAP    C$ERHRD
        046112' 000364                                                                          .WORD   244
        046114' 003611'                                                                         .WORD   CRCPAT
        046116' 013466'                                                                         .WORD   MSG12
        046120'                                  ESCAPE  TST
        046120' 104410                                                                          TRAP    C$ESCAPE
        046122' 000056                                                                          .WORD   L10162-.
                                                ;
                                                ;CHECK THE RECEIVER STATUS WORD, WHICH THE MICROCODE PLACED IN PCBB+4, FOR ANY
                                                ;ERRORS
                                                ;
        046124' 005737  000612'         40$:     TST     PCBB+4                  ;ANYTHING SET IN RECEIVER STATUS WORD 0
        046130' 001404                           BEQ     50$                     ;NO, OK
        046132'                                  ERRHRD  245,CRCPAT,MSG29        ;PRINT ERROR MESSAGE
        046132' 104456                                                                          TRAP    C$ERHRD
        046134' 000365                                                                          .WORD   245
        046136' 003611'                                                                         .WORD   CRCPAT
        046140' 015162'                                                                         .WORD   MSG29
                                                ;
                                                ;WRITE '1' TO CLEAR 'DNI'
                                                ;
```

```
046142' 004737 017362'     50$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
046146' 103004                     BCC     55$
046150'                            ERRHRD  246,CRCPAT,RACMG7       ;ERROR DNI DID NOT CLEAR!
046150' 104456                                                                     TRAP    C$ERHRD
046152' 000366                                                                     .WORD   246
046154' 003611'                                                                    .WORD   CRCPAT
046156' 012670'                                                                    .WORD   RACMG7
046160'                    55$:
046160'                      ENDSEG
046160'                                                                    10002$:
046160' 104405                                                                     TRAP    C$ESEG

                           ;
                           ;GENERATE A NEW BYTE COUNT BY SHIFTING IT OVER ONE PLACE TO THE LEFT.
                           ;THIS EFFECTIVELY MULTIPLIES THE NUMBER BY TWO.
                           ;CHECK TO SEE IF THE NEW NUMBER IS NOT TOO LARGE.
                           ;REPEAT THE TEST WITH THE NEW BYTE COUNT
                           ;
046162' 006303                     ASL     R3                      ;UP THE BYTE COUNT TO NEXT BIT POSTION
046164' 020327 002000              CMP     R3,#BIT10               ;HAVE WE CHECKED ALL BIT POSTIONS IN
                                                                   ;THE BYTE COUNT REGISTER?
046170' 001307                     BNE     28$                     ;NOT YET
046172'                      ENDSEG
046172'                                                                    10001$:
046172' 104405                                                                     TRAP    C$ESEG
                           ;
                           ;GET A NEW DATA PATTERN FROM THE LIST OF PRESELECTED DATA PATTERNS
                           ;REPEAT TEST WITH NEW DATA PATTERN
                           ;
046174' 005305                     DEC     R5                      ;HAVE WE TESTED WITH ALL DATA PATTERNS?
046176' 001277                     BNE     27$                     ;NOT YET

046200'                    ENDTST
046200'                                                                    L10162:
046200' 104401                                                                     TRAP    C$ETST
```

```
                        .SBTTL  TEST 41: RECEIVER BUFFER RECOVERY - RUNT TEST

                        ;*****************************************************************************
                        ;
                        ;THIS TEST WILL CHECK THE ABILITY OF THE RECEIVE STATE MACHINE TO REJECT A
                        ;DATAGRAM OF LESS THAN 64 BYTES AND TO RECOVER THE RECEIVER BUFFER.
                        ;
                        ;THIS TEST WILL USE MICROMODULE 'F' MICROTEST #4.
                        ;EACH TRIAL WILL CONSIST OF TWO DATAGRAM TRANSMISSIONS IN LOOPBACK MODE. EACH
                        ;TRANSMISSION WILL LOOPBACK A DATAGRAM FILLED WITH UNIQUE DATA. THE FIRST
                        ;DATAGRAM WILL BE A RUNT OF LESS THAN 64 BYTES. THE SECOND WILL BE A DATAGRAM
                        ;OF LEGAL SIZE.
                        ;
                        ;EACH TRIAL WILL START WITH THE LINK BUFFER POINTER RESET TO THE FIRST LINK
                        ;BUFFER. THE RUNT WILL BE TRANSMITTED, THEN THE VALID DATAGRAM. IF THE BUFFER
                        ;RECOVERY IS WORKING CORRECTLY, THE SECOND DATAGRAM IS EXPECTED TO BE WRITTEN
                        ;INTO THE SAME LINK MEMORY BUFFER AS WAS THE RUNT.
                        ;
                        ;THIS TEST WILL BE REPEATED WITH VARIOUS RUNT PACKET SIZES.
                        ;
                        ;THE BYTE COUNT FOR THE RUNT PACKET TRANSMISSION WILL BE PASSED VIA THE PCBB.
                        ;AFTER THE TWO TRANSMISSIONS, THE MICROCODE WILL PASS BACK THE CONTENTS OF THE
                        ;BUFFER DONE FIFO, AND THE CONTENTS OF THE FIRST DATA WORD OF THE RECEIVER
                        ;BUFFER.
                        ;
                        ;THE PCBB WILL BE FORMATTED AS FOLLOWS:
                        ;
                        ;               +-------------------------------+
                        ;PCBB+0:        !        RUNT BYTE COUNT        !
                        ;               +-------------------------------+
                        ;PCBB+2:        !   BUFFER DONE FIFO CONTENTS   !
                        ;               +-------------------------------+
                        ;PCBB+4:        !  FIRST DATA WORD OF BUFFER    !
                        ;               +-------------------------------+
                        ;
                        ;TEST SEQUENCE:
                        ;       1-LOAD MICROMODULE 'F' IF NOT ALREADY DONE SO
                        ;       2-PLACE A RUNT BYTE COUNT IN PCBB+0
                        ;       3-CLEAR PCBB+2,+4
                        ;       4-WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE
                        ;       5-SELECT MICROTEST #4
                        ;       6-WAIT FOR 'DNI'
                        ;       7-CHECK PCSR1 FOR AN ERROR CONDITION (RECEIVER INTERRUPT OCCURRED
                        ;         ON RUNT PACKET RECEPTION)
                        ;       8-CHECK PCBB+2 FOR CORRECT BUFFER DONE ADDRESS
                        ;       9-CHECK PCBB+4 FOR CORRECT DATA PATTERN
                        ;       10-WRITE '1' TO CLEAR 'DNI'
                        ;       11-REPEAT STEPS 2-10 WITH A NEW RUNT BYTE COUNT
                        ;
                        ;*****************************************************************************

046202'                 BGNTST
046202'                                                                 T41::
                        ;
                        ;CHECK TO SEE IF MODULE 'F' HAS BEEN LOADED. IF NOT LOAD IT INTO
                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
```

```
                                      ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                      ;
046202'                                 BGNSEG
046202' 104404                                                                                   TRAP    C$BSEG
046204' 022737  000106  000326'             CMP     #'F,MICRO               ;HAS MICROCODE MODULE 'F' BEEN LOADED
046212' 001004                              BNE     5$                      ;NO
046214' 122777  000001  132116              CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
046222' 001440                              BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
046224' 012737  000106  000326' 5$:         MOV     #'F,MICRO               ;GO LOAD MICRO MODULE 'F'
046232' 004737  020340'                     JSR     PC,LODMIC
046236' 103002                              BCC     10$                     ;OK
046240'                                     ESCAPE  TST
046240' 104410                                                                                   TRAP    C$ESCAPE
046242' 000370                                                                                   .WORD   L10163-.
046244' 012737  000176  000332' 10$:        MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
046252' 004737  017316'                     JSR     PC,CHKDNI
046256' 103022                              BCC     20$                     ;OK
046260' 012737  001000' 000310'             MOV     #$DNI,BITNAM
046266' 012737  001277' 000312'             MOV     #$NSET,BITSTA
046274' 012737  001342' 000314'             MOV     #$AFTER,PWHEN
046302' 012737  001357' 000316'             MOV     #$GTCMD,PCOMND
046310'                                     ERRHRD  247,RBRRUN,MSG1
046310' 104456                                                                                   TRAP    C$ERHRD
046312' 000367                                                                                   .WORD   247
046314' 003650'                                                                                  .WORD   RBRRUN
046316' 012716'                                                                                  .WORD   MSG1
046320'                                     ESCAPE  TST
046320' 104410                                                                                   TRAP    C$ESCAPE
046322' 000310                                                                                   .WORD   L10163-.
046324' 004737  017362'         20$:        JSR     PC,CLRDNI               ;CLEAR DNI BIT
046330' 103006                              BCC     25$
046332'                                     ERRHRD  248,RBRRUN,RACMG7       ;DNI DID NOT CLEAR!
046332' 104456                                                                                   TRAP    C$ERHRD
046334' 000370                                                                                   .WORD   248
046336' 003650'                                                                                  .WORD   RBRRUN
046340' 012670'                                                                                  .WORD   RACMG7
046342'                                     ESCAPE TST
046342' 104410                                                                                   TRAP    C$ESCAPE
046344' 000266                                                                                   .WORD   L10163-.
046346'                         25$:
046346'                                 ENDSEG
046346'                                                                                  10000$:
046346' 104405                                                                                   TRAP    C$ESEG
                                      ;
                                      ;R1 WILL CONTAIN THE RUNT BYTE COUNT TO BE USED BY THE MICROCODE
                                      ;
046350' 012701  000001                      MOV     #1,R1                   ;BEGIN WITH BYTE COUNT OF 1
046354'                         30$:
                                      ;
                                      ;PASS THE BYTE COUNT TO MICROCODE THROUGH PCBB+, CLEAR PCBB+2 AND PCBB+4
                                      ;
046354'                                 BGNSEG
046354' 104404                                                                                   TRAP    C$BSEG
046356' 010137  000606'                     MOV     R1,PCBB                 ;PASS BYTE COUNT TO MICROCODE
046362' 005037  000610'                     CLR     PCBB+2                  ;HERE IS WHERE THE MICRO WILL PUT THE
                                                                            ;'DONE' RECEIVE BUFFER FIFO ADDRESS
```

```
046366' 005037  000612'                  CLR     PCBB+4                  ;HERE IS WHERE THE MICRO WILL PUT THE
                                                                         ;FIRST WORD OF DATA FROM THE RECEIVE BUFFER
                                 ;
                                 ;WAIT FOR THE MICROCODE TO ENTER THE 'INMON' STATE, START MICROTEST #4 BY
                                 ;LOADING THE COMMAND FIELD OF PCSR0 WITH A 4, WAIT FOR 'DNI'
                                 ;
046372' 004737  020060'                  JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
046376' 103006                           BCC     35$                     ;OK
046400'                                  ERRHRD  249,RBRRUN,MSG46        ;PRINT ERROR
046400' 104456                                                                   TRAP    C$ERHRD
046402' 000371                                                                   .WORD   249
046404' 003650'                                                                  .WORD   RBRRUN
046406' 016666'                                                                  .WORD   MSG46
046410'                                  ESCAPE  TST                     ;LEAVE TEST
046410' 104410                                                                   TRAP    C$ESCAPE
046412' 000220                                                                   .WORD   L10163-.
046414' 012777  000004  131714  35$:     MOV     #4,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #4
046422' 012737  000275  000332'          MOV     #3*SECOND,METER         ;WAIT FOR DNI
046430' 004737  017316'                  JSR     PC,CHKDNI
046434' 103021                           BCC     40$                     ;OK
046436' 004737  020132'                  JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
046442' 103006                           BCC     36$                     ;NO, OK
046444'                                  ERRHRD  250,RBRRUN,MSG44        ;PRINT ERROR MESSAGE
046444' 104456                                                                   TRAP    C$ERHRD
046446' 000372                                                                   .WORD   250
046450' 003650'                                                                  .WORD   RBRRUN
046452' 016442'                                                                  .WORD   MSG44
046454'                                  ESCAPE  TST
046454' 104410                                                                   TRAP    C$ESCAPE
046456' 000154                                                                   .WORD   L10163-.
046460' 012702  000004          36$:     MOV     #4,R2                   ;MICROTEST #4 IS HUNG
046464'                                  ERRHRD  251,RBRRUN,MSG12        ;PRINT ERROR MESSAGE
046464' 104456                                                                   TRAP    C$ERHRD
046466' 000373                                                                   .WORD   251
046470' 003650'                                                                  .WORD   RBRRUN
046472' 013466'                                                                  .WORD   MSG12
046474'                                  ESCAPE TST
046474' 104410                                                                   TRAP    C$ESCAPE
046476' 000134                                                                   .WORD   L10163-.
                                 ;
                                 ;'DNI' SET INDICATING THE TEST IS FINISHED, NOW CHECK PCSR1 FOR AN ERROR
                                 ;CONDITION. THIS CONDITION WILL BE SET IF, AFTER THE MICROCODE TRANSMITTED THE
                                 ;RUNT PACKET, A RECEIVER INTERRUPT OCCURRED. THIS SHOULD NOT HAPPEN BECAUSE
                                 ;THE RECEIVER STATE MACHINE SHOULD GO THROUGH A BAD PACKET STATE AND NOT
                                 ;CAUSE A RECEIVER INTERRUPT
                                 ;
046500' 122777  000003  131632  40$:     CMPB    #INERR,@PCSR1           ;DID AN ERROR OCCUR?
046506' 001005                           BNE     45$                     ;NO
046510'                                  ERRHRD  252,RBRRUN,MSG30        ;YES, PRINT ERROR MESSAGE
046510' 104456                                                                   TRAP    C$ERHRD
046512' 000374                                                                   .WORD   252
046514' 003650'                                                                  .WORD   RBRRUN
046516' 015264'                                                                  .WORD   MSG30
046520' 000425                           BR      55$
                                 ;
                                 ;OK, NO RECEIVER INTERRUPT. NOW CHECK PCBB+2, WHICH IS THE CONTENTS OF THE
```

```
                                          ;BUFFER DONE FIFO AFTER THE SECOND PACKET WAS RECEIVED. BITS 14:11 SHOULD
                                          ;ALL BE ZERO INDICATING BUFFER 0 WAS RECOVERED BY THE RECEIVER STATE MACHINE.
                                          ;
046522' 042737  103777  000610'  45$:     BIC     #103777,PCBB+2          ;STRIP OFF THE FLOATING BITS
046530' 005737  000610'                   TST     PCBB+2                  ;DID DEUNA RECOVER RECEIVE BUFFER?
046534' 001407                            BEQ     50$                     ;YES
046536' 052737  100000  000610'           BIS     #100000,PCBB+2          ;NO, MAKE IT A LINK MEMORY ADDRESS
046544'                                   ERRHRD  253,RBRRUN,MSG31        ;PRINT ERROR MESSAGE
046544' 104456                                                                                    TRAP    C$ERHRD
046546' 000375                                                                                    .WORD   253
046550' 003650'                                                                                   .WORD   RBRRUN
046552' 015332'                                                                                   .WORD   MSG31
                                          ;
                                          ;NOW CHECK THE DATA THAT WAS RECEIVED INTO THE RECEIVER BUFFER. IT SHOULD
                                          ;BE AN ALTERNATING 1'S AND 0'S PATTERN.
                                          ;
046554' 022737  052525  000612'  50$:     CMP     #52525,PCBB+4           ;IS DATA GOOD IN LEGIT RECEIVE BUFFER?
046562' 001404                            BEQ     55$                     ;YES
046564'                                   ERRHRD  254,RBRRUN,MSG32        ;NO,PRINT ERROR MESSAGE
046564' 104456                                                                                    TRAP    C$ERHRD
046566' 000376                                                                                    .WORD   254
046570' 003650'                                                                                   .WORD   RBRRUN
046572' 015476'                                                                                   .WORD   MSG32
046574'                          55$:
046574'                            ENDSEG
046574'                                                                                  10001$:
046574' 104405                                                                                    TRAP    C$ESEG
                                          ;
                                          ;WRITE '1' TO CLEAR 'DNI'
                                          ;
046576' 004737  017362'                   JSR     PC,CLRDNI               ;GO CLEAR DNI
046602' 103006                            BCC     60$                     ;OK
046604'                                   ERRHRD  255,RBRRUN,RACMG7       ;ERROR DNI DID NOT CLEAR!
046604' 104456                                                                                    TRAP    C$ERHRD
046606' 000377                                                                                    .WORD   255
046610' 003650'                                                                                   .WORD   RBRRUN
046612' 012670'                                                                                   .WORD   RACMG7
046614'                                   ESCAPE TST
046614' 104410                                                                                    TRAP    C$ESCAPE
046616' 000014                                                                                    .WORD   L10163-.
                                          ;
                                          ;GENERATE A NEW RUNT PACKET BYTE COUNT. WE WILL JUST SLIDE A BIT THROUGH
                                          ;THE COUNTER, UP TO THE LAST RUNT PACKET SIZE OF 63
                                          ;
046620' 006301                   60$:     ASL     R1                      ;MOVE OVER THE ALREADY SET BITS
046622' 005201                            INC     R1                      ;SET LSB
046624' 020127  000100                    CMP     R1,#MINBYT              ;STOP WITH A BYTE COUNT GREATER THAN
                                                                          ;MINIMUM SIZE
046630' 002651                            BLT     30$
046632'                          ENDTST
046632'                                                                                  L10163:
046632' 104401                                                                                    TRAP    C$ETST
```

```
                                                .SBTTL  TEST 42: HALF-DUPLEX TEST

                                                ;*******************************************************************************
                                                ;
                                                ;THE LINK INCLUDES A 'HALF DUPLEX' MODE OF OPERATION. THIS MODE CAN BE ENABLED
                                                ;OR DISABLED THROUGH THE LINK MODE REGISTER. THE OPERATIONAL MICROCODE NORMALLY
                                                ;USES HALF-DUPLEX MODE.
                                                ;
                                                ;IN THE HALF-DUPLEX MODE, THE LINK WILL NOT RECEIVE MESSAGES ADDRESSED TO
                                                ;ITSELF. INCOMING MESSAGES LOOPED BACK WILL BE IGNORED BY THE RECEIVE STATE
                                                ;MACHINE. THE STATE MACHINE WILL NOT ISSUE A 'RECEIVER DONE' INTERRUPT AND THE
                                                ;BUFFER CAN BE RECOVERED FOR RECEIVING A LATER DATAGRAM.
                                                ;
                                                ;THIS TEST USES MICROMODULE 'F' MICROTEST #5.
                                                ;THIS TEST WILL VERIFY THE OPERATION OF HALF-DUPLEX MODE. A DATAGRAM WILL BE
                                                ;TRANSMITTED IN LOOPBACK MODE WITH THE HALF-DUPLEX BIT SET. THE MICROCODE
                                                ;WILL VERIFY THAT THE RECEIVER INTERRUPT DOES NOT OCCUR. THE MICROCODE WILL
                                                ;THEN CLEAR THE HALF-DUPLEX BIT AND LOOP A DATAGRAM AND VERIFY THAT THE
                                                ;ORIGINAL BUFFER WAS RECOVERED.
                                                ;
                                                ;THIS TEST WILL USE THE PCBB TO PASS INFORMATION. PCBB+0 WILL BE USED TO PASS
                                                ;THE CONTENTS OF THE BUFFER DONE FIFO AFTER THE SECOND DATAGRAM IS RECEIVED.
                                                ;PCBB+4 WILL BE USED TO PASS THE FIRST WORD OF DATA FROM THE RECEIVER BUFFER
                                                ;AFTER THE SECOND DATAGRAM IS TRANSMITTED.
                                                ;
                                                ;               +---------------------------+
                                                ;PCBB+0:        ! CONTENTS OF BUFFER DONE FIFO !
                                                ;               +---------------------------+
                                                ;PCBB+2:        ! FIRST DATA WORD OF BUFFER DONE!
                                                ;               +---------------------------+
                                                ;
                                                ;THE CONTENTS OF THE BUFFER DONE FIFO SHOULD BE 0 AND THE FIRST DATA WORD
                                                ;SHOULD BE AN ALTERNATING 1'S AND 0'S PATTERN.
                                                ;
                                                ;TEST SEQUENCE:
                                                ;       1-LOAD MICROMODULE 'F' IF NOT ALREADY DONE SO.
                                                ;       2-CLEAR PCBB+0 AND PCBB+2
                                                ;       3-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                                ;       4-WAIT FOR 'DNI'
                                                ;       5-CHECK PCSR1 FOR AND ERROR CONDITION (THIS SIGNIFIES THAT THE DATAGRAM
                                                ;       SENT IN HALF-DUPLEX MODE CAUSED AN INTERRUPT)
                                                ;       6-VERIFY PCBB+0 IS LINK BUFFER 0
                                                ;       7-VERIFY PCBB+2 HAS ALTERNATING 1'S AND 0'S
                                                ;
                                                ;*******************************************************************************

 046634'                                        BGNTST
 046634'                                                                                T42::
                                                ;
                                                ;CHECK TO SEE IF MODULE 'F' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                                ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                                ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                                ;
 046634'                                          BGNSEG
 046634' 104404                                                                                 TRAP    C$BSEG
 046636' 022737  000106  000326'                        CMP     #'F,MICRO               ;HAS MICROCODE MODULE 'F' BEEN LOADED
```

```
046644' 001004                                  BNE     5$                      ;NO
046646' 122777  000001  131464                  CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
046654' 001440                                  BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
046656' 012737  000106  000326' 5$:             MOV     #'F,MICRO               ;GO LOAD MICRO MODULE 'F'
046664' 004737  020340'                         JSR     PC,LODMIC
046670' 103002                                  BCC     10$                     ;OK
046672'                                         ESCAPE  TST
046672' 104410                                                                                  TRAP    C$ESCAPE
046674' 000342                                                                                  .WORD   L10164-.
046676' 012737  000176  000332' 10$:            MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
046704' 004737  017316'                         JSR     PC,CHKDNI
046710' 103022                                  BCC     20$                     ;OK
046712' 012737  001000' 000310'                 MOV     #$DNI,BITNAM
046720' 012737  001277' 000312'                 MOV     #$NSET,BITSTA
046726' 012737  001342' 000314'                 MOV     #$AFTER,PWHEN
046734' 012737  001357' 000316'                 MOV     #$GTCMD,PCOMND
046742'                                         ERRHRD  256,HAFDUP,MSG1
046742' 104456                                                                                  TRAP    C$ERHRD
046744' 000400                                                                                  .WORD   256
046746' 003723'                                                                                 .WORD   HAFDUP
046750' 012716'                                                                                 .WORD   MSG1
046752'                                         ESCAPE  TST
046752' 104410                                                                                  TRAP    C$ESCAPE
046754' 000262                                                                                  .WORD   L10164-.
046756' 004737  017362'         20$:            JSR     PC,CLRDNI               ;CLEAR DNI BIT
046762' 103006                                  BCC     25$
046764'                                         ERRHRD  257,HAFDUP,RACMG7       ;DNI DID NOT CLEAR!
046764' 104456                                                                                  TRAP    C$ERHRD
046766' 000401                                                                                  .WORD   257
046770' 003723'                                                                                 .WORD   HAFDUP
046772' 012670'                                                                                 .WORD   RACMG7
046774'                                         ESCAPE TST
046774' 104410                                                                                  TRAP    C$ESCAPE
046776' 000240                                                                                  .WORD   L10164-.
047000'                         25$:
047000'                           ENDSEG
047000'                                                                                 10000$:
047000' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;CLEAR PCBB+0 AND PCBB+2 THESE LOCATIONS ARE WHERE THE MICROCODE WILL PLACE
                                ;INFORMATION ABOUT THE TEST SUCCESS
                                ;
047002'                           BGNSEG
047002' 104404                                                                                  TRAP    C$BSEG
047004' 005037  000606'                         CLR     PCBB+0                  ;HERE IS WHERE THE MICRO WILL PUT THE
                                                                                ;'DONE' RECEIVE BUFFER FIFO ADDRESS
047010' 005037  000610'                         CLR     PCBB+2                  ;HERE IS WHERE THE MICRO WILL PUT THE
                                                                                ;FIRST WORD OF DATA FROM THE RECEIVE BUFFER
                                ;
                                ;WAIT FOR THE MICROCODE TO ENTER THE 'INMON' STATE
                                ;THEN EXECUTE MICROTEST #5 BY LOADING PCSR0 WITH A 5
                                ;WAIT FOR 'DNI'
                                ;
047014' 004737  020060'                         JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
047020' 103006                                  BCC     35$                     ;OK
047022'                                         ERRHRD  258,HAFDUP,MSG46        ;PRINT ERROR
```

```
047022' 104456                                                         TRAP    C$ERHRD
047024' 000402                                                         .WORD   258
047026' 003723'                                                        .WORD   HAFDUP
047030' 016666'                                                        .WORD   MSG46
047032'                              ESCAPE  TST              ;LEAVE TEST
047032' 104410                                                         TRAP    C$ESCAPE
047034' 000202                                                         .WORD   L10164-.
047036' 012777  000005  131272  35$:  MOV     #5,@PCSR0        ;TELL T11 TO EXECUTE MICROTEST #5
047044' 012737  000473  000332'       MOV     #5*SECOND,METER  ;WAIT FOR DNI
047052' 004737  017316'               JSR     PC,CHKDNI
047056' 103021                        BCC     40$              ;OK
047060' 004737  020132'               JSR     PC,CHKINT        ;SEE IF ANY ERROR INTERRUPTS OCCURRED
047064' 103006                        BCC     36$              ;NO, OK
047066'                               ERRHRD  259,HAFDUP,MSG44 ;PRINT ERROR MESSAGE
047066' 104456                                                         TRAP    C$ERHRD
047070' 000403                                                         .WORD   259
047072' 003723'                                                        .WORD   HAFDUP
047074' 016442'                                                        .WORD   MSG44
047076'                               ESCAPE  TST
047076' 104410                                                         TRAP    C$ESCAPE
047100' 000136                                                         .WORD   L10164-.
047102' 012702  000005          36$:  MOV     #5,R2            ;MICROTEST #5 IS HUNG
047106'                               ERRHRD  260,HAFDUP,MSG12 ;PRINT ERROR MESSAGE
047106' 104456                                                         TRAP    C$ERHRD
047110' 000404                                                         .WORD   260
047112' 003723'                                                        .WORD   HAFDUP
047114' 013466'                                                        .WORD   MSG12
047116'                               ESCAPE TST
047116' 104410                                                         TRAP    C$ESCAPE
047120' 000116                                                         .WORD   L10164-.
                                ;
                                ;AN ERROR IN PCSR1 MEANS THAT THE FIRST DATAGRAM SENT, WHICH IS THE ONE SENT
                                ;IN HALF-DUPLEX MODE, CAUSED A RECEIVER INTERRUPT.
                                ;
047122' 122777  000003  131210  40$:  CMPB    #INERR,@PCSR1    ;DID AN ERROR OCCUR?
047130' 001005                        BNE     45$              ;NO
047132'                               ERRHRD  261,HAFDUP,MSG39 ;YES, PRINT ERROR MESSAGE
047132' 104456                                                         TRAP    C$ERHRD
047134' 000405                                                         .WORD   261
047136' 003723'                                                        .WORD   HAFDUP
047140' 016056'                                                        .WORD   MSG39
047142' 000425                        BR      55$
                                ;
                                ;PCBB+0 CONTAINS THE CONTENTS OF THE BUFFER DONE FIFO AFTER THE SECOND DATAGRAM
                                ;WAS RECEIVED IT SHOULD CONTAIN THE BUFFER ADDRESS OF THE FIRST BUFFER IN LINK
                                ;MEMORY
                                ;
047144' 042737  103777  000606' 45$:  BIC     #103777,PCBB     ;STRIP OFF THE FLOATING BITS
047152' 005737  000606'               TST     PCBB             ;DID DEUNA RECOVER RECEIVE BUFFER?
047156' 001407                        BEQ     50$              ;YES
047160' 052737  100000  000606'       BIS     #100000,PCBB     ;MAKE IT A LINK MEMORY ADDRESS
047166'                               ERRHRD  262,HAFDUP,MSG40 ;NO, PRINT ERROR MESSAGE
047166' 104456                                                         TRAP    C$ERHRD
047170' 000406                                                         .WORD   262
047172' 003723'                                                        .WORD   HAFDUP
047174' 016100'                                                        .WORD   MSG40
```

```
                                    ;
                                    ;PCBB+2 CONTAINS THE FIRST DATA WORD FROM THE FIRST BUFFER IN LINK MEMORY
                                    ;IT SHOULD BE ALTERNATING 1'S AND 0'S IF THE DATA WAS PLACED IN THIS BUFFER
                                    ;
047176' 022737  052525  000610'    50$:    CMP     #52525,PCBB+2           ;IS DATA GOOD IN LEGIT RECEIVE BUFFER?
047204' 001404                             BEQ     55$                     ;YES
047206'                                    ERRHRD  263,HAFDUP,MSG41        ;NO,PRINT ERROR MESSAGE
047206' 104456                                                                                     TRAP    C$ERHRD
047210' 000407                                                                                     .WORD   263
047212' 003723'                                                                                    .WORD   HAFDUP
047214' 016176'                                                                                    .WORD   MSG41
047216'                            55$:
047216'                              ENDSEG
047216'                                                                                    10001$:
047216' 104405                                                                                     TRAP    C$ESEG
                                    ;
                                    ;WRITE '1' TO CLEAR 'DNI'
                                    ;
047220' 004737  017362'                    JSR     PC,CLRDNI               ;GO CLEAR DNI
047224' 103004                             BCC     60$                     ;OK
047226'                                    ERRHRD  264,HAFDUP,RACMG7       ;ERROR DNI DID NOT CLEAR!
047226' 104456                                                                                     TRAP    C$ERHRD
047230' 000410                                                                                     .WORD   264
047232' 003723'                                                                                    .WORD   HAFDUP
047234' 012670'                                                                                    .WORD   RACMG7
047236'                            60$:
047236'                            ENDTST
047236'                                                                                    L10164:
047236' 104401                                                                                     TRAP    C$ETST
```

```
.SBTTL  TEST 43: COLLISION TEST

;****************************************************************************
;
;THE RECEIVE STATE MACHINE REACTS TO COLLISIONS ON THE WIRE BY ACTIVATING
;THE RETRY LOGIC. THE RETRY LOGIC WAITS AN INTERVAL OF TIME BEFORE ATTEMPTING
;TO RETRANSMIT THE DATAGRAM. THE INTERVALS ARE NOT UNIFORM BUT ARE OF
;GENERALLY INCREASING PSEUDO-RANDOM DURATION. THE RETRY LOGIC WILL ATTEMPT
;TO RETRANSMIT UP TO 15 ADDITIONAL TIMES BEFORE GIVING UP.
;
;THIS TEST WILL VERIFY THAT THE RECEIVE STATE MACHINE RESPONDS TO A COLLISION
;AND THAT THE RETRY SEQUENCE IS REPORTED CORRECTLY IN THE TRANSMIT STATUS WORD.
;
;THIS TEST WILL USE MICROMODULE 'G' MICROTEST #1.
;THE LINK BOARD CONTATINS DIAGNOSTIC LOGIC THAT ALLOWS COLLSIONS TO BE SIMULATED.
;WITH THE FORCE COLLISIONS LOGIC ACTIVATED, THE RETRY HARDWARE CAN BE STEPPED
;THROUGH THE RETRY SEQUENCE. THAT IS, EVERY DATAGRAM LOOPED BACK WILL STEP
;THE RETRY LOGIC THROUGH ONE STEP OF THE RETRY SEQUENCE. THE RETRY SEQUENCE
;CAN BE VERIFIED BY CHECKING THE TRANSMIT BUFFER STATUS WORDS AFTER EACH RETRY
;STEP.
;
;THE PCBB WILL BE USED TO PASS PARAMETERS BETWEEN THE MICROCODE AND THE HOST
;PROCESSOR. PCBB+0 WILL BE USED TO PASS THE DATA TO BE LOADED INTO THE LINK
;MODE WORD. PCBB+2 WILL BE PASSED BACK BY THE MICROCODE, IT IS THE FIRST WORD
;OF THE TRANSMIT BUFFER (TRANSMIT STATUS WORD 0). PCBB+4 WILL ALSO BE PASSED
;BACK, IT IS TRANSMIT STATUS WORD 1.
;
;THE TRANSMIT STATUS WORDS SHOULD SHOW THE FOLLOWING STATUS:
;
;
;                 STATUS BITS
;
;                 WORD 0                    WORD 1
;                 -------------------       ------
;                 !                 !       !    !
;                 !                 !       !    !
;LOOPBACK         ERRS    MORE    ONE       RETRY
;  STEP #         (14)    (12)    (11)       (10)
;--------         ----    ----    ----      -----
;
; 1                0       0       1          0
;
; 2-15             0       1       0          0
;
; 16               1       0       0          1
;
;
;THE PCBB IS FORMATTED AS FOLOWS:
;
;                 +------------------------+
;PCBB+0:          !     LINK MODE WORD     !
;                 +------------------------+
;PCBB+2:          ! TRANSMIT STATUS WORD 0!
;                 +------------------------+
;PCBB+4:          ! TRANSMIT STATUS WORD 1!
;                 +------------------------+
;
```

```
                                        ;
                                        ;TEST SEQUENCE:
                                        ;       1-LOAD MICROMODULE 'G' IF NOT ALREADY SONE SO
                                        ;       2-LOAD PCBB+0 WITH PROMISCUOUS MODE, INTERNAL LOOPBACK AND FORCE
                                        ;       COLLISIONS.
                                        ;       3-WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE
                                        ;       4-EXECUTE MICROTEST #1
                                        ;       5-WAIT FOR 'DNI'
                                        ;       6-CHECK FOR 'ONE' BIT IN PCBB+0
                                        ;       7-WRITE '1' TO CLEAR 'DNI'
                                        ;       8-LOAD PCBB+0 WITH PROMISCUOUS MODE, INTERNAL LOOPBACK AND FORCE
                                        ;       COLLISIONS.
                                        ;       9-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                        ;       10-EXECUTE MICROTEST #1
                                        ;       11-WAIT FOR 'DNI'
                                        ;       12-CHECK PCBB+2 FOR 'MORE' BIT
                                        ;       13-WRITE '1' TO CLEAR 'DNI'
                                        ;       14-REPEAT STEPS 8-13 15 TIMES
                                        ;       15-LOAD PCBB+0 WITH PROMISCUOUS MODE, INTERNAL LOOPBACK AND FORCE
                                        ;       COLLSIONS
                                        ;       16-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                                        ;       17-EXECUTE MICROTEST #1
                                        ;       18-WAIT FOR 'DNI'
                                        ;       19-CHECK PCBB+4 FOR ERROR SUMMARY BIT IN PCBB+2 AND RETRY BIT IN
                                        ;       PCBB+4
                                        ;       20-WRITE '1' TO CLEAR 'DNI'
                                        ;
                                        ;**********************************************************************8

        047240'                         BGNTST
        047240'                                                                         T43::
                                        ;
                                        ;CHECK TO SEE IF MODULE 'G' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                        ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                        ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                        ;
        047240'                           BGNSEG
        047240' 104404                                                                  TRAP    C$BSEG
        047242' 022737  000107  000326'         CMP     #'G,MICRO               ;HAS MICROCODE MODULE 'G' BEEN LOADED
        047250' 001004                          BNE     5$                      ;NO
        047252' 122777  000001  131060          CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
        047260' 001440                          BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
        047262' 012737  000107  000326' 5$:     MOV     #'G,MICRO               ;GO LOAD MICRO MODULE 'G'
        047270' 004737  020340'                 JSR     PC,LODMIC
        047274' 103002                          BCC     10$                     ;OK
        047276'                                 ESCAPE  TST
        047276' 104410                                                                  TRAP    C$ESCAPE
        047300' 001026                                                                  .WORD   L10165-.
        047302' 012737  000176  000332' 10$:    MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
        047310' 004737  017316'                 JSR     PC,CHKDNI
        047314' 103022                          BCC     20$                     ;OK
        047316' 012737  001000' 000310'         MOV     #SDNI,BITNAM
        047324' 012737  001277' 000312'         MOV     #SNSET,BITSTA
        047332' 012737  001342' 000314'         MOV     #SAFTER,PWHEN
        047340' 012737  001357' 000316'         MOV     #SGTCMD,PCOMND
        047346'                                 ERRHRD  265,COLTST,MSG1
```

```
047346' 104456                                                                       TRAP    C$ERHRD
047350' 000411                                                                       .WORD   265
047352' 003753'                                                                      .WORD   COLTST
047354' 012716'                                                                      .WORD   MSG1
047356'                                 ESCAPE  TST
047356' 104410                                                                       TRAP    C$ESCAPE
047360' 000746                                                                       .WORD   L10165-.
047362' 004737  017362'         20$:    JSR     PC,CLRDNI               ;CLEAR DNI BIT
047366' 103006                          BCC     25$
047370'                                 ERRHRD  266,COLTST,RACMG7       ;DNI DID NOT CLEAR!
047370' 104456                                                                       TRAP    C$ERHRD
047372' 000412                                                                       .WORD   266
047374' 003753'                                                                      .WORD   COLTST
047376' 012670'                                                                      .WORD   RACMG7
047400'                                 ESCAPE TST
047400' 104410                                                                       TRAP    C$ESCAPE
047402' 000724                                                                       .WORD   L10165-.
047404'                         25$:
047404'                           ENDSEG
047404'                                                                     10000$:
047404' 104405                                                                       TRAP    C$ESEG

                                ;
                                ;LOAD PCSR2 WITH ADDRESS OF PORT CONTROL BLOCK
                                ;LOAD REGISTER 5 WITH BITS TO BE LOADED INTO THE LINK MODE REGISTER BY THE
                                ;MICROCODE
                                ;
047406' 012777  000606' 130726          MOV     #PCBB,@PCSR2            ;TELL DEUNA WHERE PCBB IS
047414' 005077  130724                  CLR     @PCSR3
047420' 012705  100024                  MOV     #BIT15!BIT4!BIT2,R5     ;TELL MICROCODE TO LOAD THE FOLLOWING
                                                                        ;INTO THE LINK MODE REGISTER:
                                                                        ;PROMISCUOUS MODE, INTERNAL LOOPBACK,
                                                                        ;AND FORCE COLLISIONS
                                ;
                                ;BEGIN FIRST LOOPBACK
                                ;
047424'                           BGNSEG
047424' 104404                                                                       TRAP    C$BSEG
047426' 012702  000001                  MOV     #1,R2                   ;START LOOPBACK STEP #1
047432' 010537  000606'                 MOV     R5,PCBB                 ;LOAD PCBB WITH LINK MODE REGISTER DATA
047436' 005037  000610'                 CLR     PCBB+2                  ;MICROCODE WILL PUT XMIT STAT WORD 0
047442' 005037  000612'                 CLR     PCBB+4                  ;MICROCODE WILL PUT XMIT STAT WORD 1
047446' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
047452' 103006                          BCC     30$                     ;OK
047454'                                 ERRHRD  267,COLTST,MSG46        ;PRINT ERROR
047454' 104456                                                                       TRAP    C$ERHRD
047456' 000413                                                                       .WORD   267
047460' 003753'                                                                      .WORD   COLTST
047462' 016666'                                                                      .WORD   MSG46
047464'                                 ESCAPE  TST                     ;LEAVE TEST
047464' 104410                                                                       TRAP    C$ESCAPE
047466' 000640                                                                       .WORD   L10165-.
047470' 012777  000001  130640  30$:    MOV     #1,@PCSR0               ;EXECUTE MICROTEST #1
047476' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
047504' 004737  017316'                 JSR     PC,CHKDNI
047510' 103021                          BCC     40$
047512' 004737  020132'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
```

```
047516' 103006                          BCC     35$                     ;NO, OK
047520'                                 ERRHRD  268,COLTST,MSG44        ;PRINT ERROR MESSAGE
047520' 104456                                                                                  TRAP    C$ERHRD
047522' 000414                                                                                  .WORD   268
047524' 003753'                                                                                 .WORD   COLTST
047526' 016442'                                                                                 .WORD   MSG44
047530'                                 ESCAPE  TST
047530' 104410                                                                                  TRAP    C$ESCAPE
047532' 000574                                                                                  .WORD   L10165-.
047534' 012702  000001          35$:    MOV     #1,R2                   ;MICROTEST #1 IS HUNG
047540'                                 ERRHRD  269,COLTST,MSG12
047540' 104456                                                                                  TRAP    C$ERHRD
047542' 000415                                                                                  .WORD   269
047544' 003753'                                                                                 .WORD   COLTST
047546' 013466'                                                                                 .WORD   MSG12
047550'                                 ESCAPE  TST
047550' 104410                                                                                  TRAP    C$ESCAPE
047552' 000554                                                                                  .WORD   L10165-.

                                ;
                                ;THE RESULT OF THE FIRST LOOPBACK SHOULD BE TX WORD 0 'ONE' BIT SET AND NO
                                ;OTHERS
                                ;
047554' 012703  004000          40$:    MOV     #BIT11,R3               ;'ONE' BIT SHOULD BE SET IN TX WORD 0
047560' 005004                          CLR     R4                      ;NO BITS SHOULD BE SET IN TX WORD 1
047562' 032737  040000  000610'         BIT     #BIT14,PCBB+2           ;IS 'ERROR SUMMARY' SET IN WORD 0?
047570' 001014                          BNE     45$                     ;YES, ERROR
047572' 032737  010000  000610'         BIT     #BIT12,PCBB+2           ;IS 'MORE' BIT SET IN WORD 0?
047600' 001010                          BNE     45$                     ;YES, ERROR
047602' 032737  004000  000610'         BIT     #BIT11,PCBB+2           ;IS 'ONE' BIT SET IN WORD 0?
047610' 001404                          BEQ     45$                     ;NO, ERROR
047612' 032737  002000  000612'         BIT     #BIT10,PCBB+4           ;IS 'RETRY' BIT SET IN WORD 1?
047620' 001404                          BEQ     50$                     ;NO
047622'                         45$:    ERRHRD  270,COLTST,MSG22        ;PRINT ERROR MESSAGE
047622' 104456                                                                                  TRAP    C$ERHRD
047624' 000416                                                                                  .WORD   270
047626' 003753'                                                                                 .WORD   COLTST
047630' 014326'                                                                                 .WORD   MSG22

                                ;
                                ;WRITE 'ONE' TO CLEAR 'DNI'
                                ;
047632' 004737  017362'         50$:    JSR     PC,CLRDNI               ;GO CLEAR DNI
047636' 103004                          BCC     55$                     ;OK
047640'                                 ERRHRD  271,COLTST,RACMG7       ;PRINT ERROR MESSAGE
047640' 104456                                                                                  TRAP    C$ERHRD
047642' 000417                                                                                  .WORD   271
047644' 003753'                                                                                 .WORD   COLTST
047646' 012670'                                                                                 .WORD   RACMG7
047650'                         55$:
047650'                           ENDSEG
047650'                                                                                 10001$:
047650' 104405                                                                                  TRAP    C$ESEG
                                ;
                                ;BEGIN LOOPBACKS 2-15
                                ;
047652'                           BGNSEG
```

```
047652' 104404                                                                 TRAP    C$BSEG
047654' 012702  000002                  MOV     #2,R2                   ;START LOOPS 2-15

047660' 010537  000606'         60$:    MOV     R5,PCBB                 ;LOAD PCBB WITH LINK MODE REGISTER DATA
047664' 005037  000610'                 CLR     PCBB+2                  ;MICROCODE WILL PUT XMIT STAT WORD 0
047670' 005037  000612'                 CLR     PCBB+4                  ;MICROCODE WILL PUT XMIT STAT WORD 1
047674' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
047700' 103006                          BCC     70$                     ;OK
047702'                                 ERRHRD  272,COLTST,MSG46        ;PRINT ERROR
047702' 104456                                                                 TRAP    C$ERHRD
047704' 000420                                                                 .WORD   272
047706' 003753'                                                                .WORD   COLTST
047710' 016666'                                                                .WORD   MSG46
047712'                                 ESCAPE  TST                     ;LEAVE TEST
047712' 104410                                                                 TRAP    C$ESCAPE
047714' 000412                                                                 .WORD   L10165-.
047716' 012777  000001  130412  70$:    MOV     #1,@PCSR0               ;EXECUTE MICROTEST #1
047724' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
047732' 004737  017316'                 JSR     PC,CHKDNI
047736' 103017                          BCC     80$
047740' 004737  020132'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
047744' 103004                          BCC     75$                     ;NO, OK
047746'                                 ERRHRD  273,COLTST,MSG44        ;PRINT ERROR MESSAGE
047746' 104456                                                                 TRAP    C$ERHRD
047750' 000421                                                                 .WORD   273
047752' 003753'                                                                .WORD   COLTST
047754' 016442'                                                                .WORD   MSG44
047756' 012702  000001          75$:    MOV     #1,R2                   ;MICROTEST #1 IS HUNG
047762'                                 ERRHRD  274,COLTST,MSG12
047762' 104456                                                                 TRAP    C$ERHRD
047764' 000422                                                                 .WORD   274
047766' 003753'                                                                .WORD   COLTST
047770' 013466'                                                                .WORD   MSG12
047772'                                 ESCAPE TST
047772' 104410                                                                 TRAP    C$ESCAPE
047774' 000332                                                                 .WORD   L10165-.
                                ;
                                ;THE RESULT OF LOOPBACKS 2-15 SHOULD BE THE 'MORE' BIT IN TX 0 AND NO OTHERS
                                ;
047776' 012703  010000          80$:    MOV     #BIT12,R3               ;'MORE' BIT SHOULD BE SET IN TX0
050002' 032737  040000  000610'         BIT     #BIT14,PCBB+2           ;IS 'ERROR SUMMARY' SET IN TX 0?
050010' 001014                          BNE     90$                     ;YES, ERROR
050012' 032737  010000  000610'         BIT     #BIT12,PCBB+2           ;IS 'MORE' BIT SET IN TX 0?
050020' 001410                          BEQ     90$                     ;NO, ERROR
050022' 032737  004000  000610'         BIT     #BIT11,PCBB+2           ;IS 'ONE' BIT SET IN TX 0?
050030' 001004                          BNE     90$                     ;YES, ERROR
050032' 032737  002000  000612'         BIT     #BIT10,PCBB+4           ;IS 'RETRY' BIT SET IN TX 1?
050040' 001404                          BEQ     100$                    ;NO
050042'                         90$:    ERRHRD  275,COLTST,MSG22        ;PRINT ERROR MESSAGE
050042' 104456                                                                 TRAP    C$ERHRD
050044' 000423                                                                 .WORD   275
050046' 003753'                                                                .WORD   COLTST
050050' 014326'                                                                .WORD   MSG22

                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
```

```
                                        ;
050052' 004737  017362'         100$:   JSR     PC,CLRDNI               ;GO CLEAR DNI
050056' 103004                          BCC     105$                    ;OK
050060'                                 ERRHRD  276,COLTST,RACMG7       ;PRINT ERROR MESSAGE
050060' 104456                                                                          TRAP    C$ERHRD
050062' 000424                                                                          .WORD   276
050064' 003753'                                                                         .WORD   COLTST
050066' 012670'                                                                         .WORD   RACMG7
050070' 005202                  105$:   INC     R2                      ;LOOP COUNT
050072' 022702  000020                  CMP     #16.,R2                 ;HAVE WE DONE LOOP STEPS 2-15?
050076' 001270                          BNE     60$                     ;NO CONTINUE
050100'                                   ENDSEG
050100'                                                                         10002$:
050100' 104405                                                                          TRAP    C$ESEG
                                        ;
                                        ;BEGIN LOOPBACK #16
                                        ;
050102'                                   BGNSEG
050102' 104404                                                                          TRAP    C$BSEG
050104' 010537  000606'                 MOV     R5,PCBB                 ;LOAD PCBB WITH LINK MODE REGISTER DATA
050110' 005037  000610'                 CLR     PCBB+2                  ;MICROCODE WILL PUT XMIT STAT WORD 0
050114' 005037  000612'                 CLR     PCBB+4                  ;MICROCODE WILL PUT XMIT STAT WORD 1
050120' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
050124' 103006                          BCC     110$                    ;OK
050126'                                 ERRHRD  277,COLTST,MSG46        ;PRINT ERROR
050126' 104456                                                                          TRAP    C$ERHRD
050130' 000425                                                                          .WORD   277
050132' 003753'                                                                         .WORD   COLTST
050134' 016666'                                                                         .WORD   MSG46
050136'                                 ESCAPE  TST                     ;LEAVE TEST
050136' 104410                                                                          TRAP    C$ESCAPE
050140' 000166                                                                          .WORD   L10165-.
050142' 012777  000001  130166  110$:   MOV     #1,@PCSR0               ;EXECUTE MICROTEST #1
050150' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
050156' 004737  017316'                 JSR     PC,CHKDNI
050162' 103021                          BCC     120$
050164' 004737  020132'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
050170' 103006                          BCC     115$                    ;NO, OK
050172'                                 ERRHRD  278,COLTST,MSG44        ;PRINT ERROR MESSAGE
050172' 104456                                                                          TRAP    C$ERHRD
050174' 000426                                                                          .WORD   278
050176' 003753'                                                                         .WORD   COLTST
050200' 016442'                                                                         .WORD   MSG44
050202'                                 ESCAPE  TST
050202' 104410                                                                          TRAP    C$ESCAPE
050204' 000122                                                                          .WORD   L10165-.
050206' 012702  000001          115$:   MOV     #1,R2                   ;MICROTEST #1 IS HUNG
050212'                                 ERRHRD  279,COLTST,MSG12
050212' 104456                                                                          TRAP    C$ERHRD
050214' 000427                                                                          .WORD   279
050216' 003753'                                                                         .WORD   COLTST
050220' 013466'                                                                         .WORD   MSG12
050222'                                 ESCAPE TST
050222' 104410                                                                          TRAP    C$ESCAPE
050224' 000102                                                                          .WORD   L10165-.
```

```
                                        ;
                                        ;PCBB+2 HAS TX WORD 0, IT SHOULD HAVE ERROR SUMMARY SET. PCBB+4 HAS TX WORD 1
                                        ;IT SHOULD HAVE RETRY SET
                                        ;
050226' 012703  040000                  120$:   MOV     #BI:14,R3               ;'ERROR SUMMARY' SHOULD BE SET IN TX 0
050232' 012704  002000                          MOV     #BIT10,R4               ;'RETRY' BIT SHOULD BE SET IN TX 1

050236' 032737  040000  000610'                 BIT     #BIT14,PCBB+2           ;IS 'ERROR SUMMARY' BIT SET?
050244' 001414                                  BEQ     130$                    ;NO, ERROR
050246' 032737  010000  000610'                 BIT     #BIT12,PCBB+2           ;IS 'MORE' BIT SET?
050254' 001010                                  BNE     130$                    ;YES, ERROR
050256' 032737  004000  000610'                 BIT     #BIT11,PCBB+2           ;IS 'ONE' BIT SET?
050264' 001004                                  BNE     130$                    ;YES, ERROR
050266' 032737  002000  000612'                 BIT     #BIT10,PCBB+4           ;IS 'RETRY' BIT SET?
050274' 001004                                  BNE     140$                    ;YES
050276'                                 130$:   ERRHRD  280,COLTST,MSG22        ;PRINT ERROR MESSAGE
050276' 104456                                                                                  TRAP    C$ERHRD
050300' 000430                                                                                  .WORD   280
050302' 003753'                                                                                 .WORD   COLTST
050304' 014326'                                                                                 .WORD   MSG22
                                        ;
                                        ;WRITE '1' TO CLEAR 'DNI'
                                        ;
050306' 004737  017362'                 140$:   JSR     PC,CLRDNI               ;GO CLEAR DNI BIT
050312' 103004                                  BCC     150$                    ;OK
050314'                                         ERRHRD  281,COLTST,RACMG7       ;PRINT ERROR MESSAGE
050314' 104456                                                                                  TRAP    C$ERHRD
050316' 000431                                                                                  .WORD   281
050320' 003753'                                                                                 .WORD   COLTST
050322' 012670'                                                                                 .WORD   RACMG7
050324'                                 150$:
050324'                                   ENDSEG
050324'                                                                         10003$:
050324' 104405                                                                                  TRAP    C$ESEG
050326'                                 ENDTST
050326'                                                                         L10165:
050326' 104401                                                                                  TRAP    C$ETST
```

```
                .SBTTL  TEST 44: TDR COUNTER TEST

                ;*************************************************************************
                ;
                ;THE DEUNA HAS A COUNTER DESIGNED TO HELP LOCATE FAULTS IN THE COAXIAL CABLE.
                ;THE COUNTER IS INITIALIZED WHEN A MESSAGE IS TRANSMITTED AND INCREMENTS AS
                ;THE DATAGRAM IS TRANSMITTED. COUNTING WILL STOP IF A COLLISION OCCURS OR THE
                ;CARRIER IS LOST. COUNTING ALSO STOPS IF THE 10 BIT COUNTER REACHES ITS
                ;MODULUS.
                ;
                ;THIS TEST WILL DETERMINE TAHT THE TDR COUNTER VALUE WILL CHANGE AND THAT THE
                ;COUNTER IS NOT STUCK.
                ;
                ;BECAUSE THE COUNTER COUNTS DURING TRANSMISSION OF A DATAGRAM AND WILL CONTINUE
                ;TO COUNT DURING THE TIME THAT THE TRANSMIT STATE MACHINE OPERATES, THE COUNT
                ;ACCUMULATED IN THE COUNTER DURING TRANSMISSION IS PROPORTIONAL TO THE LENGTH
                ;OF THE DATAGRAM. THIS TEST WILL USE THIS RELATION TO VERIFY THAT THE COUNTER
                ;IS NOT STUCK.
                ;
                ;THIS TEST USES MICROMODULE 'G' MICROTEST #2.
                ;THE TEST WILL SEND DATAGRAMS OVER THE LOOPBACK. THE LENGTH OF THE DATAGRAM
                ;WILL BE VARIED BY USING A INCREASING BYTE COUNT IN THE TRANSMIT BUFFER.
                ;AFTER EACH DATAGRAM HAS BEEN LOOPED BACK, THE TRANSMIT BUFFER WORD 1 WILL BE
                ;PASSED BACK TO THE HOST TO VERIFY THAT IT IS CORRECT. THE CRITERIA FOR
                ;CORRECTNESS WILL BE: INCREASING BYTE COUNTS SHOULD RESULT IN INCREASING TDR
                ;VALUES IN TRANSMIT STATUS WORD 1.
                ;
                ;THE PCBB WILL BE FORMATED AS FOLLOWS:
                ;
                ;               +----------------------+
                ;PCBB+0:        !      BYTE COUNT      !
                ;               +----------------------+
                ;PCBB+2:        ! TRANSMIT STATUS WORD 1!
                ;               +----------------------+
                ;
                ;TEST SEQUENCE:
                ;       1-LOAD MICROMODULE 'G' IF NOT ALREADY DONE SO
                ;       2-LOAD MINIMUM BYTE COUNT INTO PCBB+0
                ;       3-CLEAR PCBB+2
                ;       4-WAIT FOR THE MICROMONITOR TO ENTER THE 'INMON' STATE
                ;       5-SELECT MICROTEST #2
                ;       6-WAIT FOR 'DNI'
                ;       7-VERIFY PCBB+2 HAS NON-ZERO VALUE (TDR COUNTER NOT ZERO)
                ;       8-WRITE '1' TO CLEAR 'DNI'
                ;       9-INCREASE BYTE COUNT IN PCBB+0 BY 1
                ;       10-WAIT FOR MICROMONITOR TO ENTER 'INMON' STATE
                ;       11-SELECT MICROTEST #2
                ;       12-WAIT FOR 'DNI'
                ;       13-VERIFY VALUE IN PCBB+2 IS GREATER THAN PREVIOUS VALUE IN PCBB+2
                ;       (VERIFY THAT TDR VALUE IS GETTING LARGER WITH LARGER BYTE COUNTS)
                ;       14-WRITE '1' TO CLEAR 'DNI'
                ;       15-REPEAT STEPS 9-14 UNTIL BYTE REACHES MINIMUM SIZE +64
                ;
                ;*************************************************************************

050330'         BGNTST
```

```
050330'                                                              T44::
                                 ;
                                 ;CHECK TO SEE IF MODULE 'G' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                 ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
                                 ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                 ;
050330'                           BGNSEG
050330' 104404                                                                  TRAP    C$BSEG
050332' 022737  000107  000326'        CMP     #'G,MICRO              ;HAS MICROCODE MODULE 'G' BEEN LOADED
050340' 001004                         BNE     5$                     ;NO
050342' 122777  000001  127770         CMPB    #INMON,@PCSR1          ;YES, IS THE MICROMONITOR ACTIVE?
050350' 001440                         BEQ     20$                    ;YES SKIP LOADING THE MICROMODULE
050352' 012737  000107  000326' 5$:    MOV     #'G,MICRO              ;GO LOAD MICRO MODULE 'G'
050360' 004737  020340'                JSR     PC,LODMIC
050364' 103002                         BCC     10$                    ;OK
050366'                                ESCAPE  TST
050366' 104410                                                                  TRAP    C$ESCAPE
050370' 000460                                                                  .WORD   L10166-.
050372' 012737  000176  000332' 10$:   MOV     #2*SECOND,METER        ;WAIT FOR THE MICROMONITOR
050400' 004737  017316'                JSR     PC,CHKDNI
050404' 103022                         BCC     20$                    ;OK
050406' 012737  001000' 000310'        MOV     #SDNI,BITNAM
050414' 012737  001277' 000312'        MOV     #SNSET,BITSTA
050422' 012737  001342' 000314'        MOV     #SAFTER,PWHEN
050430' 012737  001357' 000316'        MOV     #SGTCMD,PCOMND
050436'                                ERRHRD  282,TDRCNT,MSG1
050436' 104456                                                                  TRAP    C$ERHRD
050440' 000432                                                                  .WORD   282
050442' 004001'                                                                 .WORD   TDRCNT
050444' 012716'                                                                 .WORD   MSG1
050446'                                ESCAPE  TST
050446' 104410                                                                  TRAP    C$ESCAPE
050450' 000400                                                                  .WORD   L10166-.
050452' 004737  017362'         20$:   JSR     PC,CLRDNI              ;CLEAR DNI BIT
050456' 103006                         BCC     25$
050460'                                ERRHRD  283,TDRCNT,RACMG7      ;DNI DID NOT CLEAR!
050460' 104456                                                                  TRAP    C$ERHRD
050462' 000433                                                                  .WORD   283
050464' 004001'                                                                 .WORD   TDRCNT
050466' 012670'                                                                 .WORD   RACMG7
050470'                                ESCAPE  TST
050470' 104410                                                                  TRAP    C$ESCAPE
050472' 000356                                                                  .WORD   L10166-.
050474'                          25$:
050474'                           ENDSEG
050474'                                                              10000$:
050474' 104405                                                                  TRAP    C$ESEG
                                 ;
                                 ;LOAD MINIMUM BYTE COUNT INTO PCBB+0, CLEAR PCBB+2, WAIT FOR MICROMONITOR
                                 ;EXECUTE MICROTEST #2 BY LOADING PCSR0 WITH A 2
                                 ;WAIT FOR 'DNI'
                                 ;
050476'                           BGNSEG
050476' 104404                                                                  TRAP    C$BSEG
050500' 012737  000100  000606'        MOV     #MINBYT,PCBB           ;BEGIN WITH MINIMUM BYTE COUNT
050506' 005037  000610'                CLR     PCBB+2                 ;THIS IS WHERE MICROCODE WILL PUT...
```

```
                                                                        ;THE TDR COUNTER VALUE
050512' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
050516' 103006                          BCC     35$                     ;OK
050520'                                 ERRHRD  284,TDRCNT,MSG46        ;PRINT ERROR
050520' 104456                                                                          TRAP    C$ERHRD
050522' 000434                                                                          .WORD   284
050524' 004001'                                                                         .WORD   TDRCNT
050526' 016666'                                                                         .WORD   MSG46
050530'                                 ESCAPE  TST                     ;LEAVE TEST
050530' 104410                                                                          TRAP    C$ESCAPE
050532' 000316                                                                          .WORD   L10166-.
050534' 012777  000002  127574  35$:    MOV     #2,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #2
050542' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
050550' 004737  017316'                 JSR     PC,CHKDNI
050554' 103021                          BCC     40$
050556' 004737  020132'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
050562' 103006                          BCC     36$                     ;NO, OK
050564'                                 ERRHRD  285,TDRCNT,MSG44        ;PRINT ERROR MESSAGE
050564' 104456                                                                          TRAP    C$ERHRD
050566' 000435                                                                          .WORD   285
050570' 004001'                                                                         .WORD   TDRCNT
050572' 016442'                                                                         .WORD   MSG44
050574'                                 ESCAPE  TST
050574' 104410                                                                          TRAP    C$ESCAPE
050576' 000252                                                                          .WORD   L10166-.
050600' 012702  000002          36$:    MOV     #2,R2                   ;MICROTEST #2 IS HUNG
050604'                                 ERRHRD  286,TDRCNT,MSG12
050604' 104456                                                                          TRAP    C$ERHRD
050606' 000436                                                                          .WORD   286
050610' 004001'                                                                         .WORD   TDRCNT
050612' 013466'                                                                         .WORD   MSG12
050614'                                 ESCAPE  TST
050614' 104410                                                                          TRAP    C$ESCAPE
050616' 000232                                                                          .WORD   L10166-.

                                ;
                                ;CHECK THAT THE TRANSMIT STATUS WORD 1 HAS A NON-ZERO TDR VALUE IN IT
                                ;
050620' 013701  000610'         40$:    MOV     PCBB+2,R1               ;GET MINIMUM TDR VALUE
050624' 001004                          BNE     45$                     ;SHOULD BE A NON-ZERO VALUE
050626'                                 ERRHRD  287,TDRCNT,MSG23        ;ERROR TDR COUNTER NOT COUNTING
050626' 104456                                                                          TRAP    C$ERHRD
050630' 000437                                                                          .WORD   287
050632' 004001'                                                                         .WORD   TDRCNT
050634' 014442'                                                                         .WORD   MSG23
                                ;
                                ;WRITE '1' TO CLEAR 'DNI'
                                ;
050636' 004737  017362'         45$:    JSR     PC,CLRDNI               ;GO CLEAR DNI
050642' 103004                          BCC     46$                     ;OK
050644'                                 ERRHRD  288,TDRCNT,RACMG7       ;ERROR OCCURRED PRINT ERROR MESSAGE
050644' 104456                                                                          TRAP    C$ERHRD
050646' 000440                                                                          .WORD   288
050650' 004001'                                                                         .WORD   TDRCNT
050652' 012670'                                                                         .WORD   RACMG7
050654'                         46$:
```

```
050654'                                  ENDSEG
050654'                                                                         10001$:
050654' 104405                                                                          TRAP    C$ESEG
                                 ;
                                 ;INCREASE THE BYTE COUNT
                                 ;
050656' 005237  000606'          47$:   INC     PCBB                    ;INCREASE BYTE COUNT BY 1
050662'                                  BGNSEG
050662' 104404                                                                          TRAP    C$BSEG
                                 ;
                                 ;WAIT FOR THE MICROMONITOR, EXECUTE MICROTEST #2, AND WAIT FOR 'DNI'
                                 ;
050664' 004737  020060'                 JSR     PC,CHKMON               ;WAIT FOR MICROMONITOR
050670' 103006                          BCC     50$                     ;OK
050672'                                 ERRHRD  289,TDRCNT,MSG46        ;PRINT ERROR
050672' 104456                                                                          TRAP    C$ERHRD
050674' 000441                                                                          .WORD   289
050676' 004001'                                                                         .WORD   TDRCNT
050700' 016666'                                                                         .WORD   MSG46
050702'                                 ESCAPE  TST                     ;LEAVE TEST
050702' 104410                                                                          TRAP    C$ESCAPE
050704' 000144                                                                          .WORD   L10166-.
050706' 012777  000002  127422   50$:   MOV     #2,@PCSR0               ;TELL T11 TO EXECUTE MICROTEST #2
050714' 012737  000176  000332'         MOV     #2*SECOND,METER         ;WAIT FOR DNI
050722' 004737  017316'                 JSR     PC,CHKDNI
050726' 103021                          BCC     55$
050730' 004737  020132'                 JSR     PC,CHKINT               ;SEE IF ANY ERROR INTERRUPTS OCCURRED
050734' 103006                          BCC     51$                     ;NO, OK
050736'                                 ERRHRD  290,TDRCNT,MSG44        ;PRINT ERROR MESSAGE
050736' 104456                                                                          TRAP    C$ERHRD
050740' 000442                                                                          .WORD   290
050742' 004001'                                                                         .WORD   TDRCNT
050744' 016442'                                                                         .WORD   MSG44
050746'                                 ESCAPE  TST
050746' 104410                                                                          TRAP    C$ESCAPE
050750' 000100                                                                          .WORD   L10166-.
050752' 012702  000002           51$:   MOV     #2,R2                   ;MICROTEST #2 IS HUNG
050756'                                 ERRHRD  291,TDRCNT,MSG12
050756' 104456                                                                          TRAP    C$ERHRD
050760' 000443                                                                          .WORD   291
050762' 004001'                                                                         .WORD   TDRCNT
050764' 013466'                                                                         .WORD   MSG12
050766'                                 ESCAPE  TST
050766' 104410                                                                          TRAP    C$ESCAPE
050770' 000060                                                                          .WORD   L10166-.
                                 ;
                                 ;VERIFY THAT THE TRANSMIT STATUS WORD 1 TDR VALUE IS LARGER THAN THE PREVIOUS
                                 ;VALUE
                                 ;
050772' 013702  000610'          55$:   MOV     PCBB+2,R2               ;GET NEW TDR VALUE
050776' 020201                          CMP     R2,R1                   ;IS TDR GETTING LARGER?
051000' 101006                          BHI     60$                     ;YES
051002'                                 ERRHRD  292,TDRCNT,MSG23        ;NO ERROR
051002' 104456                                                                          TRAP    C$ERHRD
051004' 000444                                                                          .WORD   292
051006' 004001'                                                                         .WORD   TDRCNT
```

```
051010' 014442'                                                              .WORD   MSG23
051012'                           ESCAPE  TST
051012' 104410                                                               TRAP    C$ESCAPE
051014' 000034                                                               .WORD   L10166-.
                         ;
                         ;WRITE '1' TO CLEAR 'DNI'
                         ;
051016' 004737 017362'   60$:     JSR     PC,CLRDNI              ;GO CLEAR DNI
051022' 103004                    BCC     65$                    ;OK
051024'                           ERRHRD  293,TDRCNT,RACMG7      ;ERROR OCCURRED PRINT ERROR MESSAGE
051024' 104456                                                               TRAP    C$ERHRD
051026' 000445                                                               .WORD   293
051030' 004001'                                                              .WORD   TDRCNT
051032' 012670'                                                              .WORD   RACMG7
051034'                  65$:
051034'                    ENDSEG
051034'                                                              10002$:
051034' 104405                                                               TRAP    C$ESEG
                         ;
                         ;CONTINUE THE LOOP UNTIL THE BYTE COUNT IS TOO LARGE FOR THE COUNTER MODULUS
                         ;
051036' 010201                    MOV     R2,R1                  ;HOLD NEW TDR VALUE FOR LATER COMPARE
051040' 023727 000610' 000164     CMP     PCBB+2,#116.           ;HAVE WE DONE ENOUGH PACKETS?
051046' 103703                    BLO     47$                    ;NO KEEP GOING
051050'                  ENDTST
051050'                                                              L10166:
051050' 104401                                                               TRAP    C$ETST
```

```
                                .SBTTL  TEST 45: RETRY LOGIC TEST

                                ;*******************************************************************************
                                ;
                                ;THE RETRY LOGIC IS ACTIVATED WHENEVER A COLLISION IS ENCOUNTERED DURING A
                                ;TRANSMISSION ATTEMPT. THE LINK STOPS TRANSMISSION AND WAITS FOR A PERIOD OF
                                ;TIME BEFORE ATTEMPTING TO RETRANSMIT.
                                ;
                                ;THE WAIT TIME IS AN INTEGRAL NUMBER OF 'SLOT TIMES'. THE NUMBER COMES FROM
                                ;A RANDOM NUMBER GENERATOR. THE NUMBER OF SLOT TIMES IS NOT EXACTLY RANDOM
                                ;SINCE THE RETRY LOGIC WAITS A GENERALLY INCREASING NUMBER OF SLOT TIMES BEFORE
                                ;TRYING TO RETRANSMIT. THIS TEST WILL VERIFY THAT THE RETRY LOGIC IS CAPABLE OF
                                ;GENERATING VARIABILITY IN THE DURATION OF THE RETRY WIAT TIMES.
                                ;
                                ;THIS TEST WILL USE MICROMODULE 'G' MICROTEST #3
                                ;THE LINK MODULE HAS A DIAGNOSTIC MAINTENANCE FACILITY MAKING IT POSSIBLE TO
                                ;SINGLE STEP THE RETRY LOGIC THROUGH THE MAXIMUM SIXTEEN RETRY STEPS. THIS
                                ;FEATURE WILL ALSO MAKE IT POSSIBLE TO MEASURE THE RETRY WAIT INTERVAL.
                                ;
                                ;THE MICROCODE WILL SET THE COLLISION BIT IN THE LINK MODE REGISTER AND
                                ;AND TRANSMIT A DATAGRAM IN LOOPBACK MODE. THE T-11 WILL COUNT WHILE WAITING
                                ;FOR THE TRANSMIT STATE MACHINE TO INTERRUPT. THE ACCUMULATED COUNT SHOULD
                                ;PROVIDE A MEASURE OF TIME TAKEN FOR THE TRANSMISSION ATTEMPT TO OCCUR. SINCE
                                ;THE COLLISION BIT IS SET, THIS INTERVAL WILL INCLUDE THE RETRY WAIT INTERVAL.
                                ;THE ACCUMULATED COUNT WILL BE WRITTEN BY THE MICROCODE TO THE PCBB.
                                ;
                                ;THE MICROTEST WILL BE EXECUTED 16 TIMES. AFTER EACH EXECUTION, THE COUNT WILL
                                ;BE READ FROM THE PCBB AND STORED IN A TABLE. THE TABLE WILL BE SCANNED TO
                                ;VERIFY THAT THEY ARE NOT ALL THE SAME.
                                ;
                                ;THE PCBB IS FORMATTED AS FOLLOWS:
                                ;
                                ;                   +-------------------------+
                                ;PCBB+0:            !       BYTE COUNT        !
                                ;                   +-------------------------+
                                ;PCBB+2:            !  TRANSMIT WAIT COUNT    !
                                ;                   +-------------------------+
                                ;
                                ;TEST SEQUENCE:
                                ;       1-LOAD MICROMODULE 'G' IF NOT ALREADY DONE SO
                                ;       2-PLACE A MINIMUM BYTE COUNT IN PCBB+0
                                ;       3-CLEAR PCBB+2
                                ;       4-WAIT FOR MICROMONITOR TO ENTER THE 'INMON' STATE
                                ;       5-SELECT MICROTEST #3
                                ;       6-WAIT FOR 'DNI'
                                ;       7-WRITE '1' TO CLEAR 'DNI'
                                ;       8-READ PCBB+2 AND PUT COUNT IN TABLE
                                ;       9-REPEAT STEPS 2-8 15 TIMES
                                ;       10-VERIFY NO 10 CONSECUTIVE ENTRIES IN THE TABLE ARE THE SAME.
                                ;*******************************************************************************

 051052'                        BGNTST
 051052'                                                                T45::
                                ;
                                ;CHECK TO SEE IF MODULE 'G' HAS BEEN LOADED. IF NOT LOAD IT INTO
                                ;THE TOP HALF OF WCS, START IT AND WAIT FOR THE MICROMONITOR TO SET 'DNI'.
```

```
                                            ;AFTER 'DNI' SETS WRITE '1' TO CLEAR IT.
                                            ;
051052'                                       BGNSEG
051052' 104404                                                                                      TRAP    C$BSEG
051054' 022737  000107  000326'                     CMP     #'G,MICRO               ;HAS MICROCODE MODULE 'G' BEEN LOADED
051062' 001004                                      BNE     5$                      ;NO
051064' 122777  000001  127246                      CMPB    #INMON,@PCSR1           ;YES, IS THE MICROMONITOR ACTIVE?
051072' 001440                                      BEQ     20$                     ;YES SKIP LOADING THE MICROMODULE
051074' 012737  000107  000326'  5$:                MOV     #'G,MICRO               ;GO LOAD MICRO MODULE 'G'
051102' 004737  020340'                             JSR     PC,LODMIC
051106' 103002                                      BCC     10$                     ;OK
051110'                                             ESCAPE  TST
051110' 104410                                                                                      TRAP    C$ESCAPE
051112' 000352                                                                                      .WORD   L10167-.
051114' 012737  000176  000332'  10$:               MOV     #2*SECOND,METER         ;WAIT FOR THE MICROMONITOR
051122' 004737  017316'                             JSR     PC,CHKDNI
051126' 103022                                      BCC     20$                     ;OK
051130' 012737  001000' 000310'                     MOV     #$DNI,BITNAM
051136' 012737  001277' 000312'                     MOV     #$NSET,BITSTA
051144' 012737  001342' 000314'                     MOV     #$AFTER,PWHEN
051152' 012737  001357' 000316'                     MOV     #$GTCMD,PCOMND
051160'                                             ERRHRD  294,RETLOG,MSG1
051160' 104456                                                                                      TRAP    C$ERHRD
051162' 000446                                                                                      .WORD   294
051164' 004031'                                                                                     .WORD   RETLOG
051166' 012716'                                                                                     .WORD   MSG1
051170'                                             ESCAPE  TST
051170' 104410                                                                                      TRAP    C$ESCAPE
051172' 000272                                                                                      .WORD   L10167-.
051174' 004737  017362'          20$:               JSR     PC,CLRDNI               ;CLEAR DNI BIT
051200' 103006                                      BCC     25$
051202'                                             ERRHRD  295,RETLOG,RACMG7       ;DNI DID NOT CLEAR!
051202' 104456                                                                                      TRAP    C$ERHRD
051204' 000447                                                                                      .WORD   295
051206' 004031'                                                                                     .WORD   RETLOG
051210' 012670'                                                                                     .WORD   RACMG7
051212'                                             ESCAPE TST
051212' 104410                                                                                      TRAP    C$ESCAPE
051214' 000250                                                                                      .WORD   L10167-.
051216'                          25$:
051216'                                       ENDSEG
051216'                                                                                 10000$:
051216' 104405                                                                                      TRAP    C$ESEG
                                            ;
                                            ;WRITE A BYTE COUT TO PCBB+0, CLEAR PCBB+2, GET POINTER TO TOP OF TABLE
                                            ;INITALIZE A COUNTER TO 16
                                            ;
051220' 012737  000100  000606'                     MOV     #MINBYT,PCBB            ;BEGIN WITH MINIMUM BYTE COUNT
051226' 005037  000610'                             CLR     PCBB+2                  ;THIS IS WHERE MICROCODE WILL PUT...
                                                                                    ;THE RETRY WAIT INTERVAL
051232' 012702  000626'                             MOV     #CNTTAB,R2              ;GET POINTER TO WAIT INTERVAL STORAGE AREA
051236' 012701  000020                              MOV     #16.,R1                 ;NUMBER OF VALUES TO STORE
                                            ;
                                            ;WAIT FOR THE MICROCODE TO ENTER THE MICROMONITOR. EXECUTE MICROTEST #3 BY
                                            ;LOADING THE COMMAND FIELD OF PCSR0 WITH A 3. WAIT FOR 'DNI'
                                            ;
```

```
051242'                                   BGNSEG
051242' 104404                                                                            TRAP    C$BSEG
051244' 004737  020060'                           JSR     PC,CHKMON          ;WAIT FOR MICROMONITOR
051250' 103006                                    BCC     35$                ;OK
051252'                                           ERRHRD  296,RETLOG,MSG46   ;PRINT ERROR
051252' 104456                                                                            TRAP    C$ERHRD
051254' 000450                                                                            .WORD   296
051256' 004031'                                                                           .WORD   RETLOG
051260' 016666'                                                                           .WORD   MSG46
051262'                                           ESCAPE  TST                ;LEAVE TEST
051262' 104410                                                                            TRAP    C$ESCAPE
051264' 000200                                                                            .WORD   L10167-.
051266' 012777  000003  127042    35$:            MOV     #3,@PCSR0          ;TELL T11 TO EXECUTE MICROTEST #3
051274' 012737  000770  000332'                   MOV     #10*SECOND,METER   ;WAIT FOR DNI
051302' 004737  017316'                           JSR     PC,CHKDNI
051306' 103021                                    BCC     40$
051310' 004737  020132'                           JSR     PC,CHKINT          ;SEE IF ANY ERROR INTERRUPTS OCCURRED
051314' 103006                                    BCC     36$                ;NO, OK
051316'                                           ERRHRD  297,RETLOG,MSG44   ;PRINT ERROR MESSAGE
051316' 104456                                                                            TRAP    C$ERHRD
051320' 000451                                                                            .WORD   297
051322' 004031'                                                                           .WORD   RETLOG
051324' 016442'                                                                           .WORD   MSG44
051326'                                           ESCAPE  TST
051326' 104410                                                                            TRAP    C$ESCAPE
051330' 000134                                                                            .WORD   L10167-.
051332' 012702  000003            36$:            MOV     #3,R2              ;MICROTEST #3 IS HUNG
051336'                                           ERRHRD  298,RETLOG,MSG12
051336' 104456                                                                            TRAP    C$ERHRD
051340' 000452                                                                            .WORD   298
051342' 004031'                                                                           .WORD   RETLOG
051344' 013466'                                                                           .WORD   MSG12
051346'                                           ESCAPE  TST
051346' 104410                                                                            TRAP    C$ESCAPE
051350' 000114                                                                            .WORD   L10167-.
                                  ;
                                  ;WRITE '1' TO CLEAR 'DNI'
                                  ;
051352' 004737  017362'           40$:            JSR     PC,CLRDNI          ;GO CLEAR DNI
051356' 103006                                    BCC     45$                ;OK
051360'                                           ERRHRD  299,RETLOG,RACMG7  ;ERROR
051360' 104456                                                                            TRAP    C$ERHRD
051362' 000453                                                                            .WORD   299
051364' 004031'                                                                           .WORD   RETLOG
051366' 012670'                                                                           .WORD   RACMG7
051370'                                           ESCAPE TST                 ;LEAVE
051370' 104410                                                                            TRAP    C$ESCAPE
051372' 000072                                                                            .WORD   L10167-.
                                  ;
                                  ;STORE THE VALUE FORM PCBB+2 IN THE TABLE AND BUMP THE POINTER TO NEXT ENTRY
                                  ;REPEAT THE TEST UNTIL ALL 16 ENTRIES ARE OBTAINED
                                  ;
051374' 013722  000610'           45$:            MOV     PCBB+2,(R2)+       ;STORE COUNTER VALUE IN THE TABLE
051400' 005301                                    DEC     R1                 ;HAVE WE STORED 16 VALUES?
051402' 001331                                    BNE     35$                ;NOT YET
                                  ;
```

```
                                ;TREAT THE TABLE AS 7 GROUPS OF 10 ENTRIES. VERIFY THAT ALL 10 ENTRIES OF EACH
                                ;GROUP ARE NOT THE SAME.
                                ;
051404' 012701  000626'                 MOV     #CNTTAB,R1                      ;POINT TO TOP OF TABLE
051410' 010102                  50$:    MOV     R1,R2                           ;GET POINTER TO FIRST ELEMENT IN THIS
                                                                                ;GROUP
051412' 010204                          MOV     R2,R4                           ;GET POINTER TO SECOND ELEMENT IN THIS
                                                                                ;GROUP
051414' 062704  000002                  ADD     #2,R4
051420' 012703  000007                  MOV     #7.,R3                          ;NUMBER OF GROUPS OF 10 TO COMPARE
051424' 022224                  55$:    CMP     (R2)+,(R4)+                     ;IS THIS PAIR THE SAME?
051426' 001010                          BNE     60$                             ;NO, START A NEW GROUP TO CHECK
051430' 005303                          DEC     R3                              ;HAVE WE CHECKED ALL 10 VALUESIN THIS
                                                                                ;GROUP?
051432' 001374                          BNE     55$                             ;NOT YET
051434'                                 ERRHRD  300,RETLOG,MSG24                ;YES, PRINT ERROR MESSAGE AND DUMP TABLE
051434' 104456                                                                                  TRAP    C$ERHRD
051436' 000454                                                                                  .WORD   300
051440' 004031'                                                                                 .WORD   RETLOG
051442' 014464'                                                                                 .WORD   MSG24
051444'                                 ESCAPE  TST
051444' 104410                                                                                  TRAP    C$ESCAPE
051446' 000016                                                                                  .WORD   L10167-.
051450' 062701  000002          60$:    ADD     #2,R1                           ;POINT TO NEXT GROUP OF 10 ELEMENTS
051454' 020127  000644'                 CMP     R1,#CNTTAB+16                   ;CHECKED ALL GROUPS OF 10?
051460' 001353                          BNE     50$                             ;NOT YET
051462'                             ENDSEG
051462'                                                                                 10001$:
051462' 104405                                                                                  TRAP    C$ESEG
051464'                         ENDTST
051464'                                                                                 L10167:
051464' 104401                                                                                  TRAP    C$ETST
```

```

                                  .SBTTL  TEST 46: PRINT DEVICE PARAMETERS TEST

                                  ;*************************************************************************
                                  ;
                                  ;       THIS TEST PRINTS THE DEFAULT PHYSICAL ADDRESS, THE MICROCODE
                                  ;       REVISION AND THE SWITCH PACK SETTINGS.
                                  ;
                                  ;       TEST SEQUENCE:
                                  ;               1.  READ DEFAULT PHYSICAL ADDRESS
                                  ;               2.  READ MICROCODE REVISION
                                  ;               3.  READ SWITCH PACK SETTINGS
                                  ;               4.  PRINT
                                  ;NOTE:
                                  ;       THIS TEST IS ONLY EXECUTED ONCE FOR EACH UNIT REGARDLESS OF THE PASS #
                                  ;
                                  ;*************************************************************************

 051466'                          BGNTST
 051466'                                                                         T46::
 051466' 005737  000674'                  TST     FRSTIM                  ; RUN THIS TEST ?
 051472' 001002                           BNE     5$                      ; YES
 051474'                                  EXIT    TST                     ; NO, EXIT
 051474' 104432                                                                          TRAP    C$EXIT
 051476' 001042                                                                          .WORD   L10170-.
                                  ;
 051500' 004737  020166'          5$:     JSR     PC,REUNA                ;GU RESET UNA
 051504' 103002                           BCC     20$                     ;OK
 051506'                                  ESCAPE  TST                     ; ABORT TEST
 051506' 104410                                                                          TRAP    C$ESCAPE
 051510' 001030                                                                          .WORD   L10170-.
                                  ;
 051512' 012777  000606' 126622   20$:    MOV     #PCBB,@PCSR2            ;LOAD PCSR2 WITH PORT CONTROL BLOCK ADR
 051520' 005077  126620                   CLR     @PCSR3                  ;LOAD PCSR3 WITH 0
 051524' 012777  000001  126604           MOV     #GETPCB,@PCSR0          ; ISSUE GET PCBB PORT COMMAND
 051532' 012737  000077  000332'          MOV     #1*SECOND,METER         ;PUT SOME TIME ON THE METER
 051540' 004737  017316'                  JSR     PC,CHKDNI               ; DNI?
 051544' 103022                           BCC     40$                     ; YES
                                                                          ;ERROR DNI FAILED TO SET!
                                                                          ;FORMAT ERROR MESSAGE
 051546' 012737  001000' 000310'          MOV     #$DNI,BITNAM
 051554' 012737  001277' 000312'          MOV     #$NSET,BITSTA
 051562' 012737  001342' 000314'          MOV     #$AFTER,PWHEN
 051570' 012737  001350' 000316'          MOV     #$GTPCB,PCOMND
 051576'                                  ERRHRD  301,PRTPAR,MSG1
 051576' 104456                                                                          TRAP    C$ERHRD
 051600' 000455                                                                          .WORD   301
 051602' 004061'                                                                         .WORD   PRTPAR
 051604' 012716'                                                                         .WORD   MSG1
 051606'                                  ESCAPE  TST                     ; AND ABORT TEST
 051606' 104410                                                                          TRAP    C$ESCAPE
 051610' 000730                                                                          .WORD   L10170-.
                                  ;
 051612' 004737  017362'          40$:    JSR     PC,CLRDNI               ; WRITE ONE TO CLEAR DNI
                                                                          ; ERROR ?
 051616' 103006                           BCC     50$                     ; NO
```

```
                                                                          ;ERROR DNI FAILED TO CLEAR!
051620'                                         ERRHRD  302,PRTPAR,RACMG7
051620' 104456                                                                                    TRAP    C$ERHRD
051622' 000456                                                                                    .WORD   302
051624' 004061'                                                                                   .WORD   PRTPAR
051626' 012670'                                                                                   .WORD   RACMG7
051630'                                         ESCAPE  TST                        ; AND ABORT TEST
051630' 104410                                                                                    TRAP    C$ESCAPE
051632' 000706                                                                                    .WORD   L10170-.
                                ;
                                ;READ DEFAULT PHYSICAL ADDRESS
                                ;
051634' 012737  000002  000606' 50$:    MOV     #RDPA,PCBB                 ;LOAD PCBB WITH READ DEFAULT PHY ADR
051642' 012737  000176  000332'         MOV     #2*SECOND,METER            ;PUT SOME TIME ON THE METER
051650' 012777  000002  126460          MOV     #GETCMD,@PCSR0             ; ISSUE GET_CMD PORT COMMAND
051656' 004737  017316'                 JSR     PC,CHKDNI                  ; DNI ?
051662' 103022                          BCC     60$                        ; YES
                                                                           ;ERROR DNI FAILED TO SET!
051664' 012737  001000' 000310'         MOV     #$DNI,BITNAM
051672' 012737  001277' 000312'         MOV     #$NSET,BITSTA
051700' 012737  001342' 000314'         MOV     #$AFTER,PWNEN
051706' 012737  001357' 000316'         MOV     #$GTCMD,PCOMND
051714'                                 ERRHRD  303,PRTPAR,MSG1 ;PRINT ERROR MESSAGE
051714' 104456                                                                                    TRAP    C$ERHRD
051716' 000457                                                                                    .WORD   303
051720' 004061'                                                                                   .WORD   PRTPAR
051722' 012716'                                                                                   .WORD   MSG1
051724'                                 ESCAPE  TST                                ; AND ABORT TEST
051724' 104410                                                                                    TRAP    C$ESCAPE
051726' 000612                                                                                    .WORD   L10170-.
                                ;
051730' 004737  017362'         60$:    JSR     PC,CLRDNI                  ; WRITE ONE TO CLEAR DNI
                                                                           ; ERROR ?
051734' 103006                          BCC     70$                        ; NO
                                                                           ;ERROR DNI FAILED TO CLEAR!
051736'                                 ERRHRD  304,PRTPAR,RACMG7          ;PRINT ERROR MESSAGE
051736' 104456                                                                                    TRAP    C$ERHRD
051740' 000460                                                                                    .WORD   304
051742' 004061'                                                                                   .WORD   PRTPAR
051744' 012670'                                                                                   .WORD   RACMG7
051746'                                 ESCAPE  TST                                ; AND ABORT TEST
051746' 104410                                                                                    TRAP    C$ESCAPE
051750' 000570                                                                                    .WORD   L10170-.
                                ;
                                ;MOVE DEFAULT PHYSICAL ADDRESS FROM PCBB -> DPA
                                ;
051752' 013737  000610' 052542' 70$:    MOV     PCBB+2,DPA
051760' 013737  000612' 052544'         MOV     PCBB+4,DPA+2
051766' 013737  000614' 052546'         MOV     PCBB+6,DPA+4
                                ;
                                ;LOAD ASCII MESSAGE (DEFADR)
                                ;
051774' 004737  017116'                 JSR     PC,HEXDPA                  ; CONVERT TO ASCII HEX
                                ;
                                ;READ MICROCODE REVISION
                                ;
```

```
052000' 012737  000016  000606' 100$:   MOV     #RPS,PCBB                 ;LOAD PCBB WITH READ PORT STATUS FUNCTION
052006' 012737  000176  000332'         MOV     #2*SECOND,METER           ;PUT SOME TIME ON THE METER
052014' 012777  000002  126314          MOV     #GETCMD,@PCSR0            ; ISSUE GET_CMD PORT COMMAND
052022' 004737  017316'                 JSR     PC,CHKDNI                 ; DNI ?
052026' 103022                          BCC     110$                      ; YES
                                                                          ;ERROR DNI FAILED TO SET
052030' 012737  001000' 000310'         MOV     #$DNI,BITNAM
052036' 012737  001277' 000312'         MOV     #$NSET,BITSTA
052044' 012737  001342' 000314'         MOV     #$AFTER,PWHEN
052052' 012737  001357' 000316'         MOV     #$GTCMD,PCOMND
052060'                                 ERRHRD  305,PRTPAR,MSG1 ;PRINT ERROR MESSAGE
052060' 104456                                                                      TRAP    C$ERHRD
052062' 000461                                                                      .WORD   305
052064' 004061'                                                                     .WORD   PRTPAR
052066' 012716'                                                                     .WORD   MSG1
052070'                                 ESCAPE  TST                       ; AND ABORT TEST
052070' 104410                                                                      TRAP    C$ESCAPE
052072' 000446                                                                      .WORD   L10170-.
                                ;
052074' 004737  017362'         110$:   JSR     PC,CLRDNI                 ; WRITE ONE TO CLEAR DNI
                                                                          ; ERROR ?
052100' 103006                          BCC     120$                      ; NO
                                                                          ;ERROR DNI FAILED TO SET
052102'                                 ERRHRD  306,PRTPAR,RACMG7         ;PRINT ERROR MESSAGE
052102' 104456                                                                      TRAP    C$ERHRD
052104' 000462                                                                      .WORD   306
052106' 004061'                                                                     .WORD   PRTPAR
052110' 012670'                                                                     .WORD   RACMG7
052112'                                 ESCAPE  TST                       ; AND ABORT TEST
052112' 104410                                                                      TRAP    C$ESCAPE
052114' 000424                                                                      .WORD   L10170-.
                                ;
                                ;MOVE MICROCODE REVISION FROM PCBB -> RREV
                                ;
052116' 013737  000610' 052550' 120$:   MOV     PCBB+2,RREV
052124' 042737  177700  052550'         BIC     #177700,RREV              ; MASK RREV
                                ;
                                ;READ SWITCH PACK
                                ;
052132' 012737  000020  000606' 130$:   MOV     #DIM,PCBB                 ;LOAD DUMP INTERNAL MEMORY FUNCTION
052140' 012737  000616' 000610'         MOV     #UDBB,PCBB+2              ;GIVE THE UNIBUS DATA BLOCK BASE ADR
052146' 005037  000612'                 CLR     PCBB+4
052152' 012737  000002  000616'         MOV     #2,UDBB                   ;SETUP TO LOAD 2 BYTES
052160' 012737  052552' 000620'         MOV     #SWPACK,UDBB+2            ; LOAD ADDRESS
052166' 005037  000622'                 CLR     UDBB+4
052172' 013737  000334' 000624'         MOV     SWADDR,UDBB+6             ; LOAD INTERNAL ADDRESS
052200' 012737  000176  000332'         MOV     #2*SECOND,METER           ;PUT SOME TIME ON THE METER
052206' 012777  000002  126122          MOV     #GETCMD,@PCSR0            ; ISSUE GET COMMAND PORT COMMAND
052214' 004737  017316'                 JSR     PC,CHKDNI                 ; DNI ?
052220' 103022                          BCC     140$                      ; YES
                                                                          ;ERROR DNI FAILED TO SET
052222' 012737  001000' 000310'         MOV     #$DNI,BITNAM
052230' 012737  001277' 000312'         MOV     #$NSET,BITSTA
052236' 012737  001342' 000314'         MOV     #$AFTER,PWHEN
052244' 012737  001357' 000316'         MOV     #$GTCMD,PCOMND
052252'                                 ERRHRD  307,PRTPAR,MSG1
```

```
052252' 104456                                                                      TRAP    C$ERHRD
052254' 000463                                                                      .WORD   307
052256' 004061'                                                                     .WORD   PRTPAR
052260' 012716'                                                                     .WORD   MSG1
052262'                                 ESCAPE  TST                 ; AND ABORT TEST
052262' 104410                                                                      TRAP    C$ESCAPE
052264' 000254                                                                      .WORD   L10170-.
                                ;
052266' 004737  017362'         140$:   JSR     PC,CLRDNI           ; WRITE ONE TO CLEAR DNI
                                                                    ; ERROR ?
052272' 103006                          BCC     150$                ; NO
052274'                                 ERRHRD  308,PRTPAR,RACMG7
052274' 104456                                                                      TRAP    C$ERHRD
052276' 000464                                                                      .WORD   308
052300' 004061'                                                                     .WORD   PRTPAR
052302' 012670'                                                                     .WORD   RACMG7
052304'                                 ESCAPE  TST                 ; AND ABORT TEST
052304' 104410                                                                      TRAP    C$ESCAPE
052306' 000232                                                                      .WORD   L10170-.
                                ;
                                ;GET SWITCH PACK INFO READY TO PRINT
                                ;
052310' 013704  052552'         150$:   MOV     SWPACK,R4           ; SWITCH PACK -> R4
052314' 042704  167777                  BIC     #167777,R4          ; MASK BIT 12
052320' 012700  000013                  MOV     #11.,R0             ; SHIFT BIT FOR INDEX
052324' 006204                  160$:   ASR     R4
052326' 005300                          DEC     R0
052330' 001375                          BNE     160$
052332' 062704  052672'                 ADD     #LPTBL,R4           ; INDEX INTO LOOP TABLE
052336' 011437  052554'                 MOV     (R4),LPMSG          ; LOAD INTO LOOP MESSAGE
052342' 013704  052552'         170$:   MOV     SWPACK,R4           ; SWITCH PACK -> R4
052346' 042704  171777                  BIC     #171777,R4          ; MASK BITS 10 AND 11
052352' 012700  000011                  MOV     #9.,R0              ; SHIFT BITS FOR INDEX
052356' 006204                  180$:   ASR     R4
052360' 005300                          DEC     R0
052362' 001375                          BNE     180$
052364' 062704  052676'                 ADD     #BTTBL,R4           ; INDEX INTO BOOT TABLE
052370' 011437  052556'                 MOV     (R4),BTMSG          ; LOAD INTO BOOT MESSAGE
                                ;
                                ; PRINT
                                ;
052374'                                 PRINTB  #FORM28,#DEFHDR     ; PRINT DEFAULT PHYSICAL ADDRESS
052374' 012746  052562'                                                             MOV     #DEFHDR,-(SP)
052400' 012746  006236'                                                             MOV     #FORM28,-(SP)
052404' 012746  000002                                                              MOV     #2,-(SP)
052410' 010600                                                                      MOV     SP,R0
052412' 104414                                                                      TRAP    C$PNTB
052414' 062706  000006                                                              ADD     #6,SP
052420'                                 PRINTB  #FORM64,RREV        ; PRINT MICROCODE REV
052420' 013746  052550'                                                             MOV     RREV,-(SP)
052424' 012746  010773'                                                             MOV     #FORM64,-(SP)
052430' 012746  000002                                                              MOV     #2,-(SP)
052434' 010600                                                                      MOV     SP,R0
052436' 104414                                                                      TRAP    C$PNTB
052440' 062706  000006                                                              ADD     #6,SP
052444'                                 PRINTB  #FORM28,#SWHDR      ; PRINT SWITCH PACK HEADER
```

```
052444' 012746 052706'                                                                  MOV     #SWHDR,-(SP)
052450' 012746 006236'                                                                  MOV     #FORM28,-(SP)
052454' 012746 000002                                                                   MOV     #2,-(SP)
052460' 010600                                                                          MOV     SP,R0
052462' 104414                                                                          TRAP    C$PNTB
052464' 062706 000006                                                                   ADD     #6,SP
052470'                                  PRINTB  #FORM28,LPMSG           ; PRINT LOOPBACK MESSAGE
052470' 013746 052554'                                                                  MOV     LPMSG,-(SP)
052474' 012746 006236'                                                                  MOV     #FORM28,-(SP)
052500' 012746 000002                                                                   MOV     #2,-(SP)
052504' 010600                                                                          MOV     SP,R0
052506' 104414                                                                          TRAP    C$PNTB
052510' 062706 000006                                                                   ADD     #6,SP
052514'                                  PRINTB  #FORM28,BTMSG           ; PRINT BOOT MESSAGE
052514' 013746 052556'                                                                  MOV     BTMSG,-(SP)
052520' 012746 006236'                                                                  MOV     #FORM28,-(SP)
052524' 012746 000002                                                                   MOV     #2,-(SP)
052530' 010600                                                                          MOV     SP,R0
052532' 104414                                                                          TRAP    C$PNTB
052534' 062706 000006                                                                   ADD     #6,SP
                            ;
052540'                     250$:
052540'                                  ENDTST
052540'                                                                     L10170:
052540' 104401                                                                          TRAP    C$ETST
```

```
                                        ;LOCAL STORAGE FOR TEST 41
 052542' 000000                         DPA::           .WORD   0               ; DEFAULT PHYSICAL ADDRESS (15:00)
 052544' 000000                                         .WORD   0               ; DEFAULT PHYSICAL ADDRESS (31:16)
 052546' 000000                                         .WORD   0               ; DEFAULT PHYSICAL ADDRESS (47:32)
                                        ;
 052550' 000000                         RREV::          .WORD   0               ; MICROCODE REVISION
                                        ;
 052552' 000000                         SWPACK::        .WORD   0               ; SWITCH PACK CONTENTS
 052554' 000000                         LPMSG::         .WORD   0               ; LOOPBACK MESSAGE ADDRESS
 052556' 000000                         BTMSG::         .WORD   0               ; BOOT MESSAGE ADDRESS
                                        ;
 052560'    000                         HEXDAT::        .BYTE   0               ; HEX DATA FOR CONVERSION
 052561'    000                         HEXVAL::        .BYTE   0               ; ASCII HEX VALUE
                                        ;
 052562' 005015  052105  042510         DEFHDR::        .ASCII  <15><12>/ETHERNET DEFAULT ADDRESS (HEX):  /
 052570' 047122  052105  042040
 052576' 043105  052501  052114
 052604' 040440  042104  042522
 052612' 051523  024040  042510
 052620' 024530  020072     040
 052625'    040     040                 DEFADR::        .ASCII  /  /
 052627'    055                                         .ASCII  /-/
 052630' 020040                                         .ASCII  /  /
 052632'    055                                         .ASCII  /-/
 052633'    040     040                                 .ASCII  /  /
 052635'    055                                         .ASCII  /-/
 052636' 020040                                         .ASCII  /  /
 052640'    055                                         .ASCII  /-/
 052641'    040     040                                 .ASCII  /  /
 052643'    055                                         .ASCII  /-/
 052644' 020040                                         .ASCII  /  /
 052646' 005015     000                                 .ASCIZ  <15><12>
                                        ;
 052651'    060                         HEXTBL::        .ASCII  /0/
 052652'    061                                         .ASCII  /1/
 052653'    062                                         .ASCII  /2/
 052654'    063                                         .ASCII  /3/
 052655'    064                                         .ASCII  /4/
 052656'    065                                         .ASCII  /5/
 052657'    066                                         .ASCII  /6/
 052660'    067                                         .ASCII  /7/
 052661'    070                                         .ASCII  /8/
 052662'    071                                         .ASCII  /9/
 052663'    101                                         .ASCII  /A/
 052664'    102                                         .ASCII  /B/
 052665'    103                                         .ASCII  /C/
 052666'    104                                         .ASCII  /D/
 052667'    105                                         .ASCII  /E/
 052670'    106                                         .ASCII  /F/
         052672'                                        .EVEN


                                        ;
                                        ;LOOP MESSAGE TABLE
 052672' 052740'                        LPTBL::         .WORD   LPMSG0
 052674' 052777'                                        .WORD   LPMSG1
                                        ;BOOT MESSAGE TABLE
```

```
052676' 053035'                          BTTBL::         .WORD   BTMSG0
052700' 053073'                                          .WORD   BTMSG1
052702' 053126'                                          .WORD   BTMSG2
052704' 053172'                                          .WORD   BTMSG3
                                         ;ASCII MESSAGES
052706' 005015  053523  052111           SWHDR::         .ASCII  <15><12>/SWITCH PACK SET FOR :/
052714' 044103  050040  041501
052722' 020113  042523  020124
052730' 047506  020122     072
052735'    015  000012                                   .ASCIZ  <15><12>
052740' 020040  020040  051440           LPMSG0::        .ASCII  /     SELF TEST LOOP DISABLED/
052746' 046105  020106  042524
052754' 052123  046040  047517
052762' 020120  044504  040523
052770' 046102  042105
052774' 005015     000                                   .ASCIZ  <15><12>
052777'    040  020040  020040           LPMSG1::        .ASCII  /     SELF TEST LOOP ENABLED/
053004' 042523  043114  052040
053012' 051505  020124  047514
053020' 050117  042440  040516
053026' 046102  042105
053032' 005015     000                                   .ASCIZ  <15><12>
053035'    040  020040  020040           BTMSG0::        .ASCII  /     NO REMOTE BOOT ENABLED/
053042' 047516  051040  046505
053050' 052117  020105  047502
053056' 052117  042440  040516
053064' 046102  042105
053070' 005015     000                                   .ASCIZ  <15><12>
053073'    040  020040  020040           BTMSG1::        .ASCII  /     REMOTE BOOT ENABLED/
053100' 042522  047515  042524
053106' 041040  047517  020124
053114' 047105  041101  042514
053122'    104
053123'    015  000012                                   .ASCIZ  <15><12>
053126' 020040  020040  051040           BTMSG2::        .ASCII  /     REMOTE BOOT ENABLED WITH ROM/
053134' 046505  052117  020105
053142' 047502  052117  042440
053150' 040516  046102  042105
053156' 053440  052111  020110
053164' 047522     115
053167'    015  000012                                   .ASCIZ  <15><12>
053172' 020040  020040  051040           BTMSG3::        .ASCII  /     REMOTE BOOT AND POWER UP BOOT BOTH ENABLED/
053200' 046505  052117  020105
053206' 047502  052117  040440
053214' 042116  050040  053517
053222' 051105  052440  020120
053230' 047502  052117  041040
053236' 052117  020110  047105
053244' 041101  042514     104
053251'    015  000012                                   .ASCIZ  <15><12>
                                                         .EVEN
```

```
                                .TITLE PARAMETER CODING
                                .SBTTL  HARDWARE PARAMETER CODING SECTION
                                ;++
                                ; THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
                                ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                ; WITH THE OPERATOR.
                                ;--
053254'                                 BGNHRD
053254' 000010                                                                                  .WORD L10171-L$HARD/2
                                                                                       L$HARD::
053256'                                 GPRMA   ASKCSR,0,0,160000,177776,NO     ;FIRST P-TABLE QUESTION
053256' 000021                                                                                  .WORD   T$CODE
053260' 053276'                                                                                 .WORD   ASKCSR
053262' 160000                                                                                  .WORD   T$LOLIM
053264' 177776                                                                                  .WORD   T$HILIM
053266'                                 GPRMA   ASKVEC,2,0,0,776,NO             ;SECOND P-TABLE QUESTION
053266' 001021                                                                                  .WORD   T$CODE
053270' 053331'                                                                                 .WORD   ASKVEC
053272' 000000                                                                                  .WORD   T$LOLIM
053274' 000776                                                                                  .WORD   T$HILIM
053276'                                 ENDHRD
                                                                                                .EVEN
053276'                                                                                L10171:
053276' 044127  052101  044440  ASKCSR: .ASCIZ  /WHAT IS THE PCSRO ADDRESS?/
053304' 020123  044124  020105
053312' 041520  051123  020060
053320' 042101  051104  051505
053326' 037523     000
053331'    127  040510  020124  ASKVEC: .ASCIZ  /WHAT IS THE VECTOR ADDRESS?/
053336' 051511  052040  042510
053344' 053040  041505  047524
053352' 020122  042101  051104
053360' 051505  037523     000
        053366'                 .EVEN
```

```
 053366' 000100                 $PATCH::.BLKW   100


         000000'                        .CSECT  MICRA
         000000'                        .CSECT  MICRB
         000000'                        .CSECT  MICRC
         000000'                        .CSECT  MICRD
         000000'                        .CSECT  MICRE
         000000'                        .CSECT  MICRF
         000000'                        .CSECT  MICRG
         000000'                        .CSECT  NOMORE
 000000'                                ENDMOD
         000001                 .END
```

```
ADR    = 000020 G         1659#
ADRERR= 000001 G          1694#   6910    7526
ASKCSR  053276R     002  13143   13156#
ASKVEC  053331R     002  13148   13161#
ASSEMB= 000010            1416
BITNAM  000310RG    002   1819#   2996    3031    3131    3299    3703    4342*   4349*   4356*   4363*   4370*   4377*   4384*
                          4386    4534*   4541    5392*   5421*   5446*   5501*   5526*   5545*   5746*   5817*   6237*   6303*
                          6434*   6477*   6538*   6664*   6711*   6759*   6838*   6985*   7115*   7191*   7192*   7230*   7231*
                          7296*   7442*   7607*   7766*   7916*   8054*   8192*   8340*   8502*   8655*   8832*   8914*   8927*
                          8936*   8948*   8957*   8966*   9059*   9228*   9401*   9566*   9755*   9918*  10157*  10380*  10535*
                         10695*  10862*  11034*  11202*  11391*  11599*  11814*  12050*  12393*  12640*  12807*  12841*  12885*
                         12930*
BITNUM  000306RG    002   1818#   2931    5064*   5081*   5167*   5184*   5201*   5218*   5235*   7159*   7189    7241*
BITSTA  000312RG    002   1820#   2930    2995    3030    4535*   4540    5063*   5166*   5393*   5422*   5439*   5492*   5525*
                          5544*   5747*   5818*   6238*   6304*   6435*   6478*   6539*   6665*   6712*   6760*   6839*   6986*
                          7116*   7157*   7297*   7443*   7608*   7767*   7917*   8055*   8193*   8341*   8503*   8656*   8833*
                          9060*   9229*   9402*   9567*   9756*   9919*  10158*  10381*  10536*  10696*  10863*  11035*  11203*
                         11392*  11600*  11815*  12051*  12394*  12641*  12808*  12842*  12886*  12931*
BIT0   = 000001 G         1632#   5590    7686
BIT00 = 000001 G          1621#   1632    1719    1733    1736    1757    1759    1761    1762    1764    5627
BIT01 = 000002 G          1620#   1631    1734    1735    1736    1757    1760    1761    1763    1764
BIT02 = 000004 G          1619#   1630    1716    1735    1757    1762    1763    1764
BIT03 = 000010 G          1618#   1629    1714    1746
BIT04 = 000020 G          1617#   1628    1712
BIT05 = 000040 G          1616#   1627    1710    1722
BIT06 = 000100 G          1615#   1626    1708    1721
BIT07 = 000200 G          1614#   1625    1706    1720    1744
BIT08 = 000400 G          1613#   1624    1718    1770
BIT09 = 001000 G          1612#   1623    1769
BIT1   = 000002 G         1631#
BIT10 = 002000 G          1611#   1715    1749    1753    7158   11510   12141   12213   12284   12292
BIT11 = 004000 G          1610#   1713    1750    1753   11285   12133   12139   12211   12290
BIT12 = 010000 G          1609#   1711    1730    1751    1753   12137   12206   12209   12288
BIT13 = 020000 G          1608#   1709    1729    1752    1753    8925    8946
BIT14 = 040000 G          1607#   1707    1728    1740    1768    5219    8955   11293   12135   12207   12283   12286
BIT15 = 100000 G          1606#   1705    1727    1739    1767    5236    7841    8934    8964   12083
BIT2   = 000004 G         1630#   7841   12083
BIT3   = 000010 G         1629#   5494
BIT4   = 000020 G         1628#   5065    5168    5438   12083
BIT5   = 000040 G         1627#   5185
BIT6   = 000100 G         1626#   1687    5202   11432
BIT7   = 000200 G         1625#
BIT8   = 000400 G         1624#   1747    1753
BIT9   = 001000 G         1623#   1748    1753    5082
BNAMT0  000360RG    002   1844#   5435    7191    7230
BNAMT1  000420RG    002   1861#   5490
BNAMT2  000460RG    002   1879#   8909
BOE    = 000400 G         1663#
BTMSG   052556RG    002  12972*  13005   13025#
BTMSG0  053035RG    002  13072   13094#
BTMSG1  053073RG    002  13073   13100#
BTMSG2  053126RG    002  13074   13106#
BTMSG3  053172RG    002  13075   13113#
BTTBL   052676RG    002  12971   13072#
CHKDNI  017316RG    002   4273#   4531    4652    4680    4879    5389    5742    5814    6234    6300    6431    6474    6536
                          6661    6708    6756    6836    6887    6982    7112    7294    7342    7440    7496    7605    7660
```

```
                              7764    7813    7914    7960    8052    8099    8190    8237    8338    8392    8500    8553    8653
                              8710    8830    8881    9057    9113    9226    9282    9399    9447    9564    9623    9753    9809
                              9916    9972   10155   10223   10378   10433   10533   10580   10693   10746   10860   10913   11032
                             11089   11200   11259   11389   11457   11597   11656   11812   11867   12048   12108   12185   12257
                             12391   12441   12508   12638   12693   12803   12838   12882   12927
CHKINT   020132RG     002     4495#   6889    7498    7662    7815    7962    8108    8246    8394    8555    8712    8883    9115
                              9284    9449    9625    9811    9974   10225   10444   10583   10748   10915   11091   11261   11459
                             11658   11869   12110   12187   12259   12443   12510   12695
CHKMON   020060RG     002     4457#   6868    7163    7330    7477    7647    7801    7948    8087    8225    8375    8536    8698
                              8865    9101    9270    9434    9611    9797    9960   10211   10421   10567   10733   10893   11072
                             11242   11445   11644   11855   12096   12173   12245   12429   12496   12681
CLEAR  = 000000 G             1674#
CLKBR    000276RG     002     1813#   4805*   4809
CLKCSR   000274RG     002     1812#   4235*   4245*   4802*
CLKFRE   000302RG     002     1815#   4807*
CLKSRV   022062RG     002     4810    4988#
CLKTAB   000274RG     002     1811#
CLKVEC   000300RG     002     1814#   4806*   4811
CLNERR   021744R      002     4882    4894#
CLRDNI   017362RG     002     4308#   4551    4665    4693    4888    5764    5854    6258    6324    6455    6498    6556    6685
                              6732    6848    6923    7046    7127    7308    7370    7454    7549    7619    7701    7778    7857
                              7926    8066    8130    8204    8268    8352    8431    8514    8592    8667    8749    8844    8975
                              9071    9161    9240    9331    9413    9481    9578    9668    9767    9849    9930   10011   10169
                             10281   10392   10463   10547   10616   10707   10786   10874   10948   11046   11123   11214   11304
                             11403   11492   11611   11724   11826   11932   12062   12152   12224   12302   12405   12476   12547
                             12652   12717   12820   12854   12898   12943
CLRINT   017410R      002     4337#   4638    5802
CNTTAB   000626RG     002     1937#   3455   12673   12738   12759
COLTST   003753RG     002     2292#  12057   12067   12101   12115   12124   12146   12157   12178   12192   12198   12218   12229
                             12250   12264   12273   12297   12307
CPUPRI   000676RG     002     1943#   4964*   7030
CRCDAT   003526RG     002     2264#  11041   11051   11077   11096   11105   11118   11128
CRCERR   003563RG     002     2269#  11209   11219   11247   11266   11275   11291   11299   11309
CRCPAT   003611RG     002     2273#  11398   11408   11450   11464   11473   11487   11497
CRCSIZ=  000004 G             1697#
CRC16    017030RG     002     4087#   6341
CRC32    016710RG     002     4015#
CSRNUM   000304RG     002     1817#   2917    2932    2946    3032    4537*   4542    5029*   5132*   5278*   5328*   5377*   5472*
                              5582*   5626*   7156*
C$AU   = 000052                1416#   4926
C$AUTO=  000061                1416#   4863
C$BRK  = 000022                1416#   4277    4462
C$BSEG=  000004                1416#   5059    5077    5162    5180    5197    5214    5231    5326    5414    5442    5468    5497
                              5518    5537    5587    5629    5645    5760    6229    6254    6288    6320    6406    6451    6468
                              6494    6520    6555    6641    6681    6701    6728    6751    6827    6867    6970    7002    7100
                              7162    7218    7282    7329    7428    7476    7593    7646    7752    7800    7905    7946    8040
                              8086    8178    8224    8326    8374    8488    8535    8641    8693    8818    8864    9045    9096
                              9214    9265    9387    9433    9552    9607    9741    9792    9904    9959   10143   10205   10366
                             10416   10521   10566   10681   10732   10785   10848   10892   11020   11068   11188   11241   11377
                             11428   11438   11585   11634   11800   11845   12036   12091   12167   12241   12379   12425   12492
                             12626   12680
C$BSUB=  000002                1416#   5024    5054    5127    5157    5382    5409    5691    5714    5736    6224    6275    6401
                              6515
C$CEFG=  000045                1416#
C$CLCK=  000062                1416#   4791
C$CLEA=  000012                1416#   4905
```

```
C$CLOS= 000035         1416#
C$CLP1= 000006         1416#    5039    5142    5288    5338
C$CVEC= 000036         1416#    5042    5098    5145    5252    5291    5299    5341    5353    7044
C$DCLN= 000044         1416#    4842    5047    5150    5296    5346
C$DODU= 000051         1416#    5045    5148    5294    5344
C$DRPT= 000024         1416#
C$DU  = 000053         1416#    4915
C$EDIT= 000003         1416#    1474
C$ERDF= 000055         1416#    5034    5137    5283    5333
C$ERHR= 000056         1416#    5068    5085    5171    5188    5205    5222    5239    5395    5424    5448    5479    5503
                        5528    5547    5595    5635    5651    5704    5726    5751    5768    5805    5822    5846    5859
                        6242    6262    6308    6328    6348    6439    6459    6482    6502    6543    6560    6577    6669
                        6689    6716    6736    6764    6774    6843    6851    6871    6892    6899    6919    6926    6990
                        7018    7034    7049    7120    7130    7166    7194    7206    7233    7301    7311    7333    7345
                        7366    7373    7447    7457    7480    7501    7510    7533    7540    7552    7612    7622    7650
                        7665    7674    7691    7704    7771    7781    7804    7818    7827    7846    7860    7921    7929
                        7951    7965    7974    7991    8059    8069    8090    8104    8111    8120    8133    8197    8207
                        8228    8242    8249    8258    8271    8345    8355    8378    8397    8406    8424    8434    8507
                        8517    8539    8558    8567    8585    8595    8660    8670    8701    8715    8724    8742    8752
                        8837    8847    8868    8886    8895    8916    8929    8938    8950    8959    8968    8978    9064
                        9074    9104    9118    9127    9150    9164    9233    9243    9273    9287    9296    9320    9334
                        9406    9416    9437    9452    9461    9474    9484    9571    9581    9614    9628    9637    9657
                        9671    9760    9770    9800    9814    9823    9838    9852    9923    9933    9963    9977    9986
                       10000   10014   10162   10172   10214   10228   10237   10254   10270   10284   10385   10395   10424
                       10447   10455   10466   10540   10550   10570   10586   10595   10609   10619   10700   10710   10736
                       10751   10760   10773   10789   10867   10877   10896   10918   10927   10941   10951   11039   11049
                       11075   11094   11103   11116   11126   11207   11217   11245   11264   11273   11289   11297   11307
                       11396   11406   11448   11462   11471   11485   11495   11604   11614   11647   11661   11670   11687
                       11702   11713   11727   11819   11829   11858   11872   11881   11895   11910   11921   11935   12055
                       12065   12099   12113   12122   12144   12155   12176   12190   12196   12216   12227   12248   12262
                       12271   12295   12305   12398   12408   12432   12446   12455   12469   12479   12499   12513   12522
                       12537   12550   12645   12655   12684   12698   12707   12720   12751   12812   12825   12846   12859
                       12890   12903   12935   12947
C$ERRO= 000060         1416#
C$ERSF= 000054         1416#
C$ERSO= 000057         1416#
C$ESCA= 000010         1416#    5400    5756    5810    5851    6247    6313    6444    6487    6548    6674    6694    6721
                        6741    6876    7023    7109    7125    7135    7171    7291    7306    7316    7338    7350    7437
                        7452    7462    7485    7506    7515    7602    7617    7627    7655    7670    7679    7709    7761
                        7776    7786    7809    7823    7832    7865    7956    7970    8049    8064    8074    8095    8116
                        8125    8187    8202    8212    8233    8254    8263    8335    8350    8360    8383    8402    84.1
                        8497    8512    8522    8544    8563    8572    8650    8665    8675    8706    8720    8729    8827
                        8842    8852    8873    8891    8900    9054    9069    9079    9109    9123    9132    9169    9223
                        9238    9248    9278    9292    9301    9339    9396    9411    9421    9442    9457    9466    9561
                        9576    9586    9619    9633    9642    9676    9750    9765    9775    9805    9819    9828    9857
                        9913    9928    9938    9968    9982    9991   10019   10152   10167   10177   10219   10233   10242
                       10375   10390   10400   10429   10452   10530   10545   10555   10575   10591   10600   10690   10705
                       10715   10741   10756   10765   10794   10857   10872   10882   10901   10923   10932   11029   11044
                       11054   11080   11099   11108   11197   11212   11222   11250   11269   11278   11386   11401   11411
                       11453   11467   11476   11594   11609   11619   11652   11666   11675   11732   11809   11824   11834
                       11863   11877   11886   12045   12060   12070   12104   12118   12127   12181   12201   12253   12267
                       12276   12388   12403   12413   12437   12451   12460   12504   12518   12527   12542   12635   12650
                       12660   12689   12703   12712   12725   12756   12796   12817   12830   12851   12864   12895   12908
                       12940   12952
C$ESEG= 000005         1416#    5075    5092    5178    5195    5212    5229    5246    5350    5431    5455    5486    5510
                        5535    5554    5602    5642    5658    5775    6252    6269    6318    6335    6449    6466    6492
```

```
                              6509    6553    6567    6679    6699    6726    6746    6781    6859    6933    6997    7041    7146
                              7213    7240    7321    7380    7467    7559    7632    7697    7791    7853    7937    8003    8079
                              8140    8217    8278    8365    8441    8527    8602    8680    8759    8857    8985    9084    9157
                              9253    9327    9426    9491    9591    9664    9780    9845    9943   10007   10182   10291   10405
                             10473   10560   10626   10720   10780   10799   10887   10958   11059   11134   11227   11314   11416
                             11502   11515   11624   11720   11839   11928   12075   12162   12236   12312   12418   12486   12557
                             12665   12763
C$ESUB= 000003                1416#   5051    5095    5154    5249    5405    5557    5711    5733    5778    6272    6355    6512
                              6586
C$ETST= 000001                1416#   5101    5255    5303    5356    5560    5607    5661    5781    5866    6358    6597    6790
                              6936    7056    7247    7383    7562    7715    7874    8006    8143    8281    8444    8605    8764
                              8988    9180    9350    9494    9689    9872   10033   10294   10483   10629   10808   10961   11142
                             11323   11525   11745   11942   12315   12566   12766   13015
C$EXIT= 000032                1416#   4891   12790
C$GETB= 000026                1416#
C$GETW= 000027                1416#
C$GMAN= 000043                1416#
C$GPHR= 000042                1416#   4823
C$GPLO= 000030                1416#
C$GPRI= 000040                1416#
C$INIT= 000011                1416#   4846
C$INLP= 000020                1416#
C$MANI= 000050                1416#
C$MEM = 000031                1416#   4785
C$MSG = 000023                1416#   2925    2941    2962    2975    2988    3005    3024    3041    3054    3075    3101    3114
                              3126    3139    3151    3176    3189    3209    3223    3238    3266    3294    3308    3337    3364
                              3391    3426    3438    3476    3524    3543    3563    3583    3611    3630    3671    3698    3711
                              3731    3758    3770    3782    3794    3806    3833    3853    3895    3907    3946    3970    3982
C$OPEN= 000034                1416#
C$PNTB= 000014                1416#   2921    2936    2950    2958    2970    2983    3000    3013    3020    3036    3049    3062
                              3071    3083    3090    3097    3109    3122    3135    3147    3159    3165    3172    3185    3198
                              3205    3219    3234    3247    3254    3262    3275    3282    3290    3304    3317    3325    3333
                              3345    3352    3360    3372    3379    3387    3401    3411    3421    3434    3446    3452    3462
                              3469    3486    3493    3499    3506    3513    3520    3532    3539    3551    3559    3571    3579
                              3593    3600    3607    3619    3626    3638    3645    3652    3659    3667    3679    3687    3694
                              3707    3720    3727    3740    3746    3754    3766    3778    3790    3802    3814    3821    3829
                              3841    3849    3862    3869    3876    3883    3891    3903    3917    3925    3933    3941    3954
                              3960    3966    3978   12981   12988   12995   13002   13009
C$PNTF= 000017                1416#   4390    4546    4557    4579    4645    4660    4673    4688    4701    4799    4885
C$PNTS= 000016                1416#
C$PNTX= 000015                1416#   4427
C$QIO = 000377                1416#
C$RDBU= 000007                1416#
C$REFG= 000047                1416#   4755    4760    4775    4780
C$RESE= 000033                1416#
C$REVI= 000003                1416#   1473
C$RFLA= 000021                1416#
C$RPT = 000025                1416#   4722
C$SEFG= 000046                1416#
C$SPRI= 000041                1416#   4234    4248    7009    7182
C$SVEC= 000037                1416#   4813    5020    5123    5275    5323    6976    7142
C$TPRI= 000013                1416#
DATALD  001774RG       002    2106#   6441    6461    6484    6504    6545    6562    6579
DATERR= 000000 G              1693#   6906    7522
DBFRAM  002531RG       002    2169#   8347    8357    8380    8399    8408    8426    8436
DEFADR  052625RG       002    4141   13036#
```

```
DEFHDR  052562RG    002  12977    13030#
DFPTBL  000262RG    002   1584#
DIAGMC= 000000            1416
DIM   = 000020 G          1791#    6280     6533    12918
DMABLK  002406RG    002   2153#    7923     7931     7953     7967     7976     7993
DMAFRM  002336RG    002   2146#    7773     7783     7806     7820     7829     7848     7862
DMATO   002270RG    002   2139#    7614     7624     7652     7667     7676     7693     7706
DNI   = 004000 G          1713#    4274     4309     4365     4368     5462     5718     5723
DNIB  = 000010 G          1714#    4308     4367
DPA     052542RG    002   4142    12869*   12870*   12871*   13017#
DPPAT   002625RG    002   2181#    8662     8672     8703     8717     8726     8744     8754
DWFRAM  002567RG    002   2175#    8509     8519     8541     8560     8569     8587     8597
EF.CON= 000036 G          1639#    4754
EF.NEW= 000035 G          1640#    4774
EF.PWR= 000034 G          1641#    4759
EF.RES= 000037 G          1638#
EF.STA= 000040 G          1637#    4779
ENP   = 000400 G          1770#
ERRINT  000670RG    002   1940#    3911     3919     3927     3935     4495*    4496
ERRS  = 040000 G          1768#
EVL   = 000004 G          1657#
ESEND = 002100            1416#
ESLOAD= 000[illegible]35  1416#    1497
FATI  = 000400 G          1718#    4379     4382
FATIB = 000001 G          1719#    4381
FIFTST  003073RG    002   2213#    9573     9583     9616     9630     9639     9659     9673
FORM1   004141RG    002   2315#    2918
FORM10  004715RG    002   2379#    3046
FORM11  005001RG    002   2388#    3059     3080     3342     3369
FORM12  005030RG    002   2392#    3068     3231
FORM13  005123RG    002   2402#    3017
FORM15  005174RG    002   2409#    3087
FORM16  005250RG    002   2417#    3094
FORM17  005326RG    002   2425#    3106
FORM18  005411RG    002   2434#    3119
FORM19  005467RG    002   2442#    3132
FORM2   004176RG    002   2320#    2933
FORM20  005544RG    002   2450#    3144
FORM21  005604RG    002   2456#    3156
FORM22  005656RG    002   2463#    3162
FORM23  005736RG    002   2471#    3169
FORM24  006006RG    002   2478#    3182
FORM25  006040RG    002   2483#    3195
FORM26  006113RG    002   2491#    3202
FORM27  006166RG    002   2499#    3216     3259     3287     3357     3384     3684     3846     3888
FORM28  006236RG    002   2506#    3244     3724    12978    12992    12999    13006
FORM29  006243RG    002   2507#    3251     3376
FORM3   004234RG    002   2325#    2947
FORM30  006266RG    002   2511#    3272
FORM31  006316RG    002   2515#    3279
FORM32  006344RG    002   2519#    3301
FORM33  006441RG    002   2530#    3314     3873
FORM34  006477RG    002   2536#    3322
FORM35  006562RG    002   2545#    3330
FORM36  006662RG    002   2556#    3349     3880
FORM37  006720RG    002   2561#    3398
```

FORM38	007031RG	002	2574#	3408	3418										
FORM39	007122RG	002	2584#	3431											
FORM4	004303RG	002	2332#	2955											
FORM40	007163RG	002	2590#	3443											
FORM41	007246RG	002	2599#	3449											
FORM42	007314RG	002	2606#	3459											
FORM43	007323RG	002	2608#	3466											
FORM44	007331RG	002	2610#	3496											
FORM45	007410RG	002	2618#	3503											
FORM46	007436RG	002	2622#	3510											
FORM47	007465RG	002	2626#	3517											
FORM48	007514RG	002	2630#	3529											
FORM49	007611RG	002	2641#	3536											
FORM5	004352RG	002	2339#	2967											
FORM50	007665RG	002	2649#	3556	3576	3590									
FORM51	007754RG	002	2659#	3548											
FORM52	010006RG	002	2664#	3568											
FORM53	010044RG	002	2669#	3597											
FORM54	010116RG	002	2676#	3604											
FORM55	010144RG	002	2680#	3616											
FORM56	010215RG	002	2687#	3635	3811										
FORM57	010270RG	002	2695#	3623	3642	3691									
FORM58	010340RG	002	2702#	3649											
FORM59	010416RG	002	2710#	3656	3737	3818									
FORM6	004415RG	002	2345#	2997											
FORM60	010511RG	002	2720#	3664	3743	3826									
FORM61	010576RG	002	2729#	3676	3838										
FORM62	010661RG	002	2738#	3704	4387										
FORM63	010734RG	002	2746#	3717											
FORM64	010773RG	002	2752#	12985											
FORM65	011044RG	002	2759#												
FORM66	011072RG	002	2763#	3763											
FORM67	011157RG	002	2772#	3775											
FORM68	011243RG	002	2781#	3787											
FORM69	011340RG	002	2792#	3799											
FORM7	004473RG	002	2353#	3010											
FORM70	011456RG	002	2806#	3859											
FORM71	011520RG	002	2812#	3866											
FORM72	011562RG	002	2818#	3900	3938										
FORM73	011634RG	002	2825#	3914											
FORM74	011677RG	002	2831#	3922											
FORM75	011754RG	002	2839#	3930											
FORM76	012022RG	002	2846#	3483											
FORM77	012056RG	002	2851#	3490											
FORM78	012132RG	002	2859#	3751											
FORM79	012172RG	002	2865#	3951											
FORM8	004555RG	002	2362#	2980	4554										
FORM80	012241RG	002	2872#	3957											
FORM81	012320RG	002	2880#	3963											
FORM82	012363RG	002	2886#	3975											
FORM83	012453RG	002	2896#	4424											
FORM9	004637RG	002	2371#	3033	4543										
FREMEM	000324RG	002	1826#	4787*	4788*	6289	6296	6339	6411	6419	6525	6531	6571	7947	10187
			10193	10250	10251	10252	10262								
FRESIZ	000322RG	002	1825#	4786*	4787										
FRSTIM	000674RG	002	1942#	4783*	4816*	4840*	12787								

```
F$AU   = 000015     1416#   4920    4925
F$AUTO= 000020      1416#   4860    4862
F$BGN  = 000040     1416#   1420    2915    2928    2944    2965    2978    2991    3008    3027    3044    3057    3078
                    3104    3117    3129    3142    3154    3179    3192    3212    3226    3241    3269    3297    3311
                    3340    3367    3394    3429    3441    3479    3527    3546    3566    3586    3614    3633    3574
                    3701    3714    3734    3761    3773    3785    3797    3809    3836    3856    3898    3910    3949
                    3973    4716    4720    4733    4749    4860    4872    4891    4909    4913    4920    4924    4943
                    4962    4988    5014    5023    5050    5053    5059    5077    5094    5100    5117    5126    5153
                    5156    5162    5180    5197    5214    5231    5248    5254    5269    5302    5317    5326    5355
                    5376    5381    5400    5404    5408    5414    5442    5468    5497    5518    5537    5556    5559
                    5581    5587    5606    5625    5629    5645    5660    5688    5690    5710    5713    5732    5735
                    5756    5760    5777    5780    5801    5810    5851    5865    6221    6223    6229    6247    6254
                    6271    6274    6288    6313    6320    6354    6357    6397    6400    6406    6444    6451    6468
                    6487    6494    6511    6514    6520    6548    6555    6585    6596    6636    6641    6674    6681
                    6694    6701    6721    6728    6741    6751    6789    6820    6827    6867    6876    6935    6964
                    6970    7002    7023    7055    7093    7100    7109    7125    7135    7162    7171    7218    7246
                    7275    7282    7291    7306    7316    7329    7338    7350    7382    7421    7428    7437    7452
                    7462    7476    7485    7506    7515    7561    7586    7593    7602    7617    7627    7646    7655
                    7670    7679    7709    7714    7745    7752    7761    7776    7786    7800    7809    7823    7832
                    7865    7873    7898    7905    7946    7956    7970    8005    8033    8040    8049    8064    8074
                    8086    8095    8116    8125    8142    8171    8178    8187    8202    8212    8224    8233    8254
                    8263    8280    8319    8326    8335    8350    8360    8374    8383    8402    8411    8443    8481
                    8488    8497    8512    8522    8535    8544    8563    8572    8604    8634    8641    8650    8665
                    8675    8693    8706    8720    8729    8763    8811    8818    8827    8842    8852    8864    8873
                    8891    8900    8987    9038    9045    9054    9069    9079    9096    9109    9123    9132    9169
                    9179    9207    9214    9223    9238    9248    9265    9278    9292    9301    9339    9349    9380
                    9387    9396    9411    9421    9433    9442    9457    9466    9493    9545    9552    9561    9576
                    9586    9607    9619    9633    9642    9676    9688    9734    9741    9750    9765    9775    9792
                    9805    9819    9828    9857    9871    9897    9904    9913    9928    9938    9959    9968    9982
                    9991    10019   10032   10136   10143   10152   10167   10177   10205   10219   10233   10242   10293
                    10359   10366   10375   10390   10400   10416   10429   10452   10482   10514   10521   10530   10545
                    10555   10566   10575   10591   10600   10628   10674   10681   10690   10705   10715   10732   10741
                    10756   10765   10785   10794   10807   10841   10848   10857   10872   10882   10892   10901   10923
                    10932   10960   11013   11020   11029   11044   11054   11068   11080   11099   11108   11141   11181
                    11188   11197   11212   11222   11241   11250   11269   11278   11322   11370   11377   11386   11401
                    11411   11428   11438   11453   11467   11476   11524   11578   11585   11594   11609   11619   11634
                    11652   11666   11675   11732   11744   11793   11800   11809   11824   11834   11845   11863   11877
                    11886   11941   12029   12036   12045   12060   12070   12091   12104   12118   12127   12167   12181
                    12201   12241   12253   12267   12276   12314   12372   12379   12388   12403   12413   12425   12437
                    12451   12460   12492   12504   12518   12527   12542   12565   12619   12626   12635   12650   12660
                    12680   12689   12703   12712   12725   12756   12765   12786   12790   12796   12817   12830   12851
                    12864   12895   12908   12940   12952   13014   13139   13179
F$CLEA= 000007      1416#   4872    4904
F$DU   = 000016     1416#   4909    4914
F$END  = 000041     1416#   1420    2926    2942    2963    2976    2989    3006    3025    3042    3055    3076    3102
                    3115    3127    3140    3152    3177    3190    3210    3224    3239    3267    3295    3309    3338
                    3365    3392    3427    3439    3477    3525    3544    3564    3584    3612    3631    3672    3699
                    3712    3732    3759    3771    3783    3795    3807    3834    3854    3896    3908    3947    3971
                    3983    4718    4723    4847    4864    4891    4906    4911    4916    4922    4927    4948    4968
                    4996    5014    5023    5050    5052    5053    5076    5093    5094    5096    5100    5102    5117
                    5126    5153    5155    5156    5179    5196    5213    5230    5247    5248    5250    5254    5256
                    5269    5302    5304    5317    5351    5355    5357    5376    5381    5400    5404    5406    5408
                    5432    5456    5487    5511    5536    5555    5556    5558    5559    5561    5581    5603    5606
                    5608    5625    5643    5659    5660    5662    5688    5690    5710    5712    5713    5732    5734
                    5735    5756    5776    5777    5779    5780    5782    5801    5810    5851    5865    5867    6221
                    6223    6247    6253    6270    6271    6273    6274    6313    6319    6336    6354    6356    6357
```

```
                     6359   6397   6400   6444   6450   6467   6487   6493   6510   6511   6513   6514   6548
                     6554   6568   6585   6587   6596   6598   6636   6674   6680   6694   6700   6721   6727
                     6741   6747   6782   6789   6791   6820   6860   6876   6934   6935   6937   6964   6998
                     7023   7042   7055   7057   7093   7109   7125   7135   7147   7171   7214   7241   7246
                     7248   7275   7291   7306   7316   7322   7338   7350   7381   7382   7384   7421   7437
                     7452   7462   7468   7485   7506   7515   7560   7561   7563   7586   7602   7617   7627
                     7633   7655   7670   7679   7698   7709   7714   7716   7745   7761   7776   7786   7792
                     7809   7823   7832   7854   7865   7873   7875   7898   7938   7956   7970   8004   8005
                     8007   8033   8049   8064   8074   8080   8095   8116   8125   8141   8142   8144   8171
                     8187   8202   8212   8218   8233   8254   8263   8279   8280   8282   8319   8335   8350
                     8360   8366   8383   8402   8411   8442   8443   8445   8481   8497   8512   8522   8528
                     8544   8563   8572   8603   8604   8606   8634   8650   8665   8675   8681   8706   8720
                     8729   8760   8763   8765   8811   8827   8842   8852   8858   8873   8891   8900   8986
                     8987   8989   9038   9054   9069   9079   9085   9109   9123   9132   9158   9169   9179
                     9181   9207   9223   9238   9248   9254   9278   9292   9301   9328   9339   9349   9351
                     9380   9396   9411   9421   9427   9442   9457   9466   9492   9493   9495   9545   9561
                     9576   9586   9592   9619   9633   9642   9665   9676   9688   9690   9734   9750   9765
                     9775   9781   9805   9819   9828   9846   9857   9871   9873   9897   9913   9928   9938
                     9944   9968   9982   9991  10008  10019  10032  10034  10136  10152  10167  10177  10183
                    10219  10233  10242  10292  10293  10295  10359  10375  10390  10400  10406  10429  10452
                    10474  10482  10484  10514  10530  10545  10555  10561  10575  10591  10600  10627  10628
                    10630  10674  10690  10705  10715  10721  10741  10756  10765  10781  10794  10800  10807
                    10809  10841  10857  10872  10882  10888  10901  10923  10932  10959  10960  10962  11013
                    11029  11044  11054  11060  11080  11099  11108  11135  11141  11143  11181  11197  11212
                    11222  11228  11250  11269  11278  11315  11322  11324  11370  11386  11401  11411  11417
                    11453  11467  11476  11516  11524  11526  11578  11594  11609  11619  11625  11652  11666
                    11675  11721  11732  11744  11746  11793  11809  11824  11834  11840  11863  11877  11886
                    11929  11941  11943  12029  12045  12060  12070  12076  12104  12118  12127  12163  12181
                    12201  12237  12253  12267  12276  12313  12314  12316  12372  12388  12403  12413  12419
                    12437  12451  12460  12487  12504  12518  12527  12542  12558  12565  12567  12619  12635
                    12650  12660  12666  12689  12703  12712  12725  12756  12764  12765  12767  12786  12790
                    12796  12817  12830  12851  12864  12895  12908  12940  12952  13014  13016  13154  13179
F$HARD= 000004       1416#  13139  13152
F$HW  = 000013       1416#   1582   1590
F$INIT= 000006       1416#   4749   4845
F$JMP = 000050       1416#   4718   4891   4911   4922  12790
F$MOD = 000000       1416#   1420  13179
F$MSG = 000011       1416#   2915   2924   2928   2940   2944   2961   2965   2974   2978   2987   2991   3004
                     3008   3023   3027   3040   3044   3053   3057   3074   3078   3100   3104   3113   3117
                     3125   3129   3138   3142   3150   3154   3175   3179   3188   3192   3208   3212   3222
                     3226   3237   3241   3265   3269   3293   3297   3307   3311   3336   3340   3363   3367
                     3390   3394   3425   3429   3437   3441   3475   3479   3523   3527   3542   3546   3562
                     3566   3582   3586   3610   3614   3629   3633   3670   3674   3697   3701   3710   3714
                     3730   3734   3757   3761   3769   3773   3781   3785   3793   3797   3805   3809   3832
                     3836   3852   3856   3894   3898   3906   3910   3945   3949   3969   3973   3981
F$PROT= 000021       1416#   4733   4740
F$PWR = 000017       1416#
F$RPT = 000012       1416#   4716   4721
F$SEG = 000003       1416#   5059   5074   5077   5091   5162   5177   5180   5194   5197   5211   5214   5228
                     5231   5245   5326   5349   5414   5430   5442   5454   5468   5485   5497   5509   5518
                     5534   5537   5553   5587   5601   5629   5641   5645   5657   5760   5774   6229   6251
                     6254   6268   6288   6317   6320   6334   6406   6448   6451   6465   6468   6491   6494
                     6508   6520   6552   6555   6566   6641   6678   6681   6698   6701   6725   6728   6745
                     6751   6780   6827   6858   6867   6932   6970   6996   7002   7040   7100   7145   7162
                     7212   7218   7239   7282   7320   7329   7379   7428   7466   7476   7558   7593   7631
                     7646   7696   7752   7790   7800   7852   7905   7936   7946   8002   8040   8078   8086
```

```
                              8139    8178    8216    8224    8277    8326    8364    8374    8440    8488    8526    8535    8601
                              8641    8679    8693    8758    8818    8856    8864    8984    9045    9083    9096    9156    9214
                              9252    9265    9326    9387    9425    9433    9490    9552    9590    9607    9663    9741    9779
                              9792    9844    9904    9942    9959   10006   10143   10181   10205   10290   10366   10404   10416
                             10472   10521   10559   10566   10625   10681   10719   10732   10779   10785   10798   10848   10886
                             10892   10957   11020   11058   11068   11133   11188   11226   11241   11313   11377   11415   11428
                             11438   11501   11514   11585   11623   11634   11719   11800   11838   11845   11927   12036   12074
                             12091   12161   12167   12235   12241   12311   12379   12417   12425   12485   12492   12556   12626
                             12664   12680   12762
F$SOFT= 000005                1416#
F$SRV = 000010                1416#   4943    4946    4962    4966    4988    4994
F$SUB = 000002                1416#   5024    5050    5054    5094    5127    5153    5157    5248    5382    5404    5409    5556
                              5691    5710    5714    5732    5736    5777    6224    6271    6275    6354    6401    6511    6515
                              6585
F$SW  = 000014                1416#
F$TEST= 000001                1416#   5015    5100    5118    5254    5270    5302    5318    5355    5377    5559    5582    5606
                              5626    5660    5689    5780    5802    5865    6222    6357    6398    6596    6637    6789    6821
                              6935    6965    7055    7094    7246    7276    7382    7422    7561    7587    7714    7746    7873
                              7899    8005    8034    8142    8172    8280    8320    8443    8482    8604    8635    8763    8812
                              8987    9039    9179    9208    9349    9381    9493    9546    9688    9735    9871    9898   10032
                             10137   10293   10360   10482   10515   10628   10675   10807   10842   10960   11014   11141   11182
                             11322   11371   11524   11579   11744   11794   11941   12030   12314   12373   12565   12620   12765
                             12787   13014
GETCMD= 000002 G              1734#   4678    4708    6299    6472    6534    6706    6754   12837   12881   12926
GETCRC  016752R      002      4023    4046#
GETPCB= 000001 G              1733#   4650    6232    6429    6659   12801
G$CNTO= 000200                1416#
G$DELM= 000372                1416#
G$DISP= 000003                1416#
G$EXCP= 000400                1416#
G$HILI= 000002                1416#
G$LOLI= 000001                1416#
G$ND  = 000000                1416#  13142   13147
G$OFFS= 000400                1416#  13142   13147
G$OFSI= 000376                1416#  13142   13147
G$PRMA= 000001                1416#  13142   13147
G$PRMD= 000002                1416#
G$PRML= 000000                1416#
G$RADA= 000140                1416#
G$RADB= 000000                1416#
G$RADD= 000040                1416#
G$RADL= 000120                1416#
G$RADO= 000020                1416#  13142   13147
G$XFER= 000004                1416#
G$YES = 000010                1416#
HAFDUP  003723RG     002      2287#  11821   11831   11860   11874   11883   11897   11912   11923   11937
HELP  = 000000                1391#   1416    1427    1515    1572    1586    1592#   1595    4713#   4741    4751    4849    4999#
                             13124#  13137   13155
HEXDAT  052560RG     002      4144*   4179    4215   13027#
HEXDPA  017116RG     002      4135#  12875
HEXH    017200RG     002      4145    4176#
HEXL    017236RG     002      4147    4212#
HEXTBL  052651RG     002      4187    4218   13049#
HEXVAL  052561RG     002      4146    4148    4188*   4219*  13028#
HOE   = 100000 G              1670#
IBE   = 010000 G              1667#
```

```
ICAB   = 040000 G            1740#   5522
IDU    = 000040 G            1660#
IE     = 000100 G            1687#   4235    6979    6980    7176    7178
IER    = 020000 G            1668#
INERR  = 000003              1700#   3480    6904    7520    8417    8578    8735    9471    9835    9997    10248   10606   10770
                             10938   11113   11684   11892
INITH  = 177777 G            1688#   4020
INITL  = 177777 G            1689#   4021
INMON  = 000001              1698#   4459    6830    7103    7285    7431    7596    7755    7908    8043    8181    8329    8491
                             8644    8821    9048    9217    9390    9555    9744    9907    10146   10369   10524   10684   10851
                             11023   11191   11380   11588   11803   12039   12382   12629
INTBIT   002156RG    002     2126#   7122    7132    7168    7196    7208    7235
INTE   = 000100 G            1721#   5697    5701
INTR   = 000200 G            1720#   5418
INTST  = 000002              1699#   6770
INTVEC   002121RG    002     2121#   6992    7020    7036    7051
IOADR  = 020000 G            1684#   1685
IOSIZ  = 020000 G            1680#   1685
ISR    = 000100 G            1661#
IXE    = 004000 G            1666#
I$AU   = 000041              1416#   4920#   4927#
I$AUTO = 000041              1416#   4860#   4864#
I$CLN  = 000041              1416#   4872#   4891    4906#
I$DU   = 000041              1416#   4909#   4916#
I$HRD  = 000041              13139#  13154#
I$INIT = 000041              1416#   4749#   4847#
I$MOD  = 000041              1416#   1420#   13179#
I$MSG  = 000041              1416#   2915#   2926#   2928#   2942#   2944#   2963#   2965#   2976#   2978#   2989#   2991#   3006#
                             3008#   3025#   3027#   3042#   3044#   3055#   3057#   3076#   3078#   3102#   3104#   3115#   3117#
                             3127#   3129#   3140#   3142#   3152#   3154#   3177#   3179#   3190#   3192#   3210#   3212#   3224#
                             3226#   3239#   3241#   3267#   3269#   3295#   3297#   3309#   3311#   3338#   3340#   3365#   3367#
                             3392#   3394#   3427#   3429#   3439#   3441#   3477#   3479#   3525#   3527#   3544#   3546#   3564#
                             3566#   3584#   3586#   3612#   3614#   3631#   3633#   3672#   3674#   3699#   3701#   3712#   3714#
                             3732#   3734#   3759#   3761#   3771#   3773#   3783#   3785#   3795#   3797#   3807#   3809#   3834#
                             3836#   3854#   3856#   3896#   3898#   3908#   3910#   3947#   3949#   3971#   3973#   3983#
I$PROT = 000040              1416#   4733#
I$PTAB = 000041              1416#
I$PWR  = 000041              1416#
I$RPT  = 000041              1416#   4716#   4723#
I$SEG  = 000041              1416#   5014    5023    5053    5059#   5076#   5077#   5093#   5117    5126    5156    5162#   5179#
                             5180#   5196#   5197#   5213#   5214#   5230#   5231#   5247#   5269    5317    5326#   5351#   5376
                             5381    5408    5414#   5432#   5442#   5456#   5468#   5487#   5497#   5511#   5518#   5536#   5537#
                             5555#   5581    5587#   5603#   5625    5629#   5643#   5645#   5659#   5688    5690    5713    5735
                             5760#   5776#   5801    6221    6223    6229#   6253#   6254#   6270#   6274    6288#   6319#   6320#
                             6336#   6397    6400    6406#   6450#   6451#   6467#   6468#   6493#   6494#   6510#   6514    6520#
                             6554#   6555#   6568#   6636    6641#   6680#   6681#   6700#   6701#   6727#   6728#   6747#   6751#
                             6782#   6820    6827#   6860#   6867#   6934#   6964    6970#   6998#   7002#   7042#   7093    7100#
                             7147#   7162#   7214#   7218#   7241#   7275    7282#   7322#   7329#   7381#   7421    7428#   7468#
                             7476#   7560#   7586    7593#   7633#   7646#   7698#   7745    7752#   7792#   7800#   7854#   7898
                             7905#   7938#   7946#   8004#   8033    8040#   8080#   8086#   8141#   8171    8178#   8218#   8224#
                             8279#   8319    8326#   8366#   8374#   8442#   8481    8488#   8528#   8535#   8603#   8634    8641#
                             8681#   8693#   8760#   8811    8818#   8858#   8864#   8986#   9038    9045#   9085#   9096#   9158#
                             9207    9214#   9254#   9265#   9328#   9380    9387#   9427#   9433#   9492#   9545    9552#   9592#
                             9607#   9665#   9734    9741#   9781#   9792#   9846#   9897    9904#   9944#   9959#   10008#  10136
                             10143#  10183#  10205#  10292#  10359   10366#  10406#  10416#  10474#  10514   10521#  10561#  10566#
                             10627#  10674   10681#  10721#  10732#  10781#  10785#  10800#  10841   10848#  10888#  10892#  10959#
```

```
                                11013   11020#  11060#  11068#  11135#  11181   11188#  11228#  11241#  11315#  11370   11377#  11417#
                                11428#  11438#  11516#  11578   11585#  11625#  11634#  11721#  11793   11800#  11840#  11845#  11929#
                                12029   12036#  12076#  12091#  12163#  12167#  12237#  12241#  12313#  12372   12379#  12419#  12425#
                                12487#  12492#  12558#  12619   12626#  12666#  12680#  12764#  12786
ISSETU= 000041                   1416#
ISSRV = 000041                   1416#   4943#   4948#   4962#   4968#   4988#   4996#
ISSUB = 000041                   1416#   5014    5023#   5050#   5052#   5053#   5094#   5096#   5117    5126#   5153#   5155#   5156#
                                 5248#   5250#   5269    5317    5376    5381#   5404#   5406#   5408#   5556#   5558#   5581    5625
                                 5688    5690#   5710#   5712#   5713#   5732#   5734#   5735#   5777#   5779#   5801    6221    6223#
                                 6271#   6273#   6274#   6354#   6356#   6397    6400#   6511#   6513#   6514#   6585#   6587#   6636
                                 6820    6964    7093    7275    7421    7586    7745    7898    8033    8171    8319    8481    8634
                                 8811    9038    9207    9380    9545    9734    9897    10136   10359   10514   10674   10841   11013
                                11181   11370   11578   11793   12029   12372   12619   12786
ISTST = 000041                   1416#   5014#   5023    5053    5100#   5102#   5117#   5126    5156    5254#   5256#   5269#   5302#
                                 5304#   5317#   5355#   5357#   5376#   5381    5400    5408    5559#   5561#   5581#   5606#   5608#
                                 5625#   5660#   5662#   5688#   5690    5713    5735    5756    5780#   5782#   5801#   5810    5851
                                 5865#   5867#   6221#   6223    6247    6274    6313    6357#   6359#   6397#   6400    6444    6487
                                 6514    6548    6596#   6598#   6636#   6674    6694    6721    6741    6789#   6791#   6820#   6876
                                 6935#   6937#   6964#   7023    7055#   7057#   7093#   7109    7125    7135    7171    7246#   7248#
                                 7275#   7291    7306    7316    7338    7350    7382#   7384#   7421#   7437    7452    7462    7485
                                 7506    7515    7561#   7563#   7586#   7602    7617    7627    7655    7670    7679    7709    7714#
                                 7716#   7745#   7761    7776    7786    7809    7823    7832    7865    7873#   7875#   7898#   7956
                                 7970    8005#   8007#   8033#   8049    8064    8074    8095    8116    8125    8142#   8144#   8171#
                                 8187    8202    8212    8233    8254    8263    8280#   8282#   8319#   8335    8350    8360    8383
                                 8402    8411    8443#   8445#   8481#   8497    8512    8522    8544    8563    8572    8604#   8606#
                                 8634#   8650    8665    8675    8706    8720    8729    8763#   8765#   8811#   8827    8842    8852
                                 8873    8891    8900    8987#   8989#   9038#   9054    9069    9079    9109    9123    9132    9169
                                 9179#   9181#   9207#   9223    9238    9248    9278    9292    9301    9339    9349#   9351#   9380#
                                 9396    9411    9421    9442    9457    9466    9493#   9495#   9545#   9561    9576    9586    9619
                                 9633    9642    9676    9688#   9690#   9734#   9750    9765    9775    9805    9819    9828    9857
                                 9871#   9873#   9897#   9913    9928    9938    9968    9982    9991    10019   10032#  10034#  10136#
                                10152   10167   10177   10219   10233   10242   10293#  10295#  10359#  10375   10390   10400   10429
                                10452   10482#  10484#  10514#  10530   10545   10555   10575   10591   10600   10628#  10630#  10674#
                                10690   10705   10715   10741   10756   10765   10794   10807#  10809#  10841#  10857   10872   10882
                                10901   10923   10932   10960#  10962#  11013#  11029   11044   11054   11080   11099   11108   11141#
                                11143#  11181#  11197   11212   11222   11250   11269   11278   11322#  11324#  11370#  11386   11401
                                11411   11453   11467   11476   11524#  11526#  11578#  11594   11609   11619   11652   11666   11675
                                11732   11744#  11746#  11793#  11809   11824   11834   11863   11877   11886   11941#  11943#  12029#
                                12045   12060   12070   12104   12118   12127   12181   12201   12253   12267   12276   12314#  12316#
                                12372#  12388   12403   12413   12437   12451   12460   12504   12518   12527   12542   12565#  12567#
                                12619#  12635   12650   12660   12689   12703   12712   12725   12756   12765#  12767#  12786#  12790
                                12796   12817   12830   12851   12864   12895   12908   12940   12952   13014#  13016#
JSJMP = 000167                   1416#   4718    4911    4922
LASFT   002026RG        002      2111#   6671    6691    6718    6738    6766    6776
LASM  = 000001 6                 1776#   4706    6752
LIM   = 000021 6                 1792#   4632    6423    6653
LINADR= 100000 6                 1686#   9597    9682    9786    9788    9865    9949    9951    10026
LINSIZ= 077774 6                 1682#
LNKARB  003241RG        002      2231#   10164   10174   10216   10230   10239   10256   10272   10286
LNKBYT  002727RG        002      2194#   9066    9076    9106    9120    9129    9152    9166
LNKMEM  002240RG        002      2135#   7449    7459    7482    7503    7512    7535    7542    7554
LODMIC  020340R         002      4571#   6833    7106    7288    7434    7599    7758    7911    8046    8184    8332    8494    8647
                                 8824    9051    9220    9393    9558    9747    9910    10149   10372   10527   10687   10854   11026
                                11194   11383   11591   11806   12042   12385   12632
LOE   = 040000 6                 1669#
LOT   = 000010 6                 1658#
```

```
LPMSG   052554RG   002   12964*   12998    13024#
LPMSGO  052740RG   002   13069    13082#
LPMSG1  052777RG   002   13070    13088#
LPTBL   052672RG   002   12963    13069#
L$ACP   000110RG   002   1504#
L$APT   000036RG   002   1462#
L$AU    022026RG   002   1489     4920#
L$AUT   000070RG   002   1488#
L$AUTO  021652RG   002   1505     4860#
L$CCP   000106RG   002   1502#
L$CLEA  021654RG   002   1503     4872#
L$CO    000032RG   002   1458#
L$DEPO  000011RG   002   1440#
L$DESC  000706RG   002   1495     1970#
L$DESP  000076RG   002   1494#
L$DEVP  000060RG   002   1480#
L$DISP  000124RG   002   1465     1524#
L$DLY   000116RG   002   1510#
L$DTP   000040RG   002   1464#
L$DTYP  000034RG   002   1460#
L$DU    022020RG   002   1491     4909#
L$DUT   000072RG   002   1490#
L$DVTY  000700RG   002   1481     1962#
L$EF    000052RG   002   1475#
L$ENVI  000044RG   002   1468#
L$ETP   000102RG   002   1498#
L$EXP1  000046RG   002   1470#
L$EXP4  000064RG   002   1484#
L$EXP5  000066RG   002   1486#
L$HARD  053256RG   002   1447     13139    13140#
L$HIME  000120RG   002   1512#
L$HPCP  000016RG   002   1446#
L$HPTP  000022RG   002   1450#
L$HW    000262RG   002   1451     1582     1583#
L$ICP   000104RG   002   1500#
L$INIT  021212RG   002   1501     4749#
L$LADP  000026RG   002   1454#
L$LAST= ****** GX        1455
L$LOAD  000100RG   002   1496#
L$LUN   000074RG   002   1492#
L$MREV  000050RG   002   1472#
L$NAME  000000RG   002   1429#
L$PRIO  000042RG   002   1466#
L$PROT  021204RG   002   1507     4733#
L$PRT   000112RG   002   1506#
L$REPP  000062RG   002   1482#
L$REV   000010RG   002   1438#
L$RPT   021176RG   002   1483     4716#
L$SPC   000056RG   002   1478#
L$SPCP  000020RG   002   1448#
L$SPTP  000024RG   002   1452#
L$STA   000030RG   002   1456#
L$TEST  000114RG   002   1508#
L$TIML  000014RG   002   1444#
L$UNIT  000012RG   002   1442#    4819
L10000  000266R    002   1582     1590#
```

L10001	012524R	002	2924#	
L10002	012566R	002	2940#	
L10003	012640R	002	2961#	
L10004	012666R	002	2974#	
L10005	012714R	002	2987#	
L10006	012762R	002	3004#	
L10007	013026R	002	3023#	
L10010	013074R	002	3040#	
L10011	013122R	002	3053#	
L10012	013172R	002	3074#	
L10013	013260R	002	3100#	
L10014	013306R	002	3113#	
L10015	013330R	002	3125#	
L10016	013356R	002	3138#	
L10017	013400R	002	3150#	
L10020	013464R	002	3175#	
L10021	013510R	002	3188#	
L10022	013556R	002	3208#	
L10023	013604R	002	3222#	
L10024	013634R	002	3237#	
L10025	013726R	002	3265#	
L10026	014020R	002	3293#	
L10027	014050R	002	3307#	
L10030	014144R	002	3336#	
L10031	014234R	002	3363#	
L10032	014324R	002	3390#	
L10033	014440R	002	3425#	
L10034	014462R	002	3437#	
L10035	014612R	002	3475#	
L10036	015000R	002	3523#	
L10037	015044R	002	3542#	
L10040	015112R	002	3562#	
L10041	015160R	002	3582#	
L10042	015262R	002	3610#	
L10043	015330R	002	3629#	
L10044	015474R	002	3670#	
L10045	015572R	002	3697#	
L10046	015620R	002	3710#	
L10047	015670R	002	3730#	
L10050	015766R	002	3757#	
L10051	016010R	002	3769#	
L10052	016032R	002	3781#	
L10053	016054R	002	3793#	
L10054	016076R	002	3805#	
L10055	016174R	002	3832#	
L10056	016246R	002	3852#	
L10057	016416R	002	3894#	
L10060	016440R	002	3906#	
L10061	016602R	002	3945#	
L10062	016664R	002	3969#	
L10063	016706R	002	3981#	
L10064	021202R	002	4719	4721#
L10066	021606R	002	4845#	
L10067	021652R	002	4862#	
L10070	022016R	002	4892	4904#
L10071	022024R	002	4912	4914#

L10072	022032R	002	4923	4925#						
L10073	022042R	002	4946#							
L10074	022060R	002	4966#							
L10075	022074R	002	4994#							
L10076	022306R	002	5100#							
L10077	022200R	002	5050#							
L10100	022276R	002	5094#							
L10101	022640R	002	5254#							
L10102	022414R	002	5153#							
L10103	022630R	002	5248#							
L10104	022752R	002	5302#							
L10105	023070R	002	5355#							
L10106	023556R	002	5401	5559#						
L10107	023160R	002	5404#							
L10110	023554R	002	5556#							
L10111	023640R	002	5606#							
L10112	023736R	002	5660#							
L10113	024160R	002	5757	5780#						
L10114	024002R	002	5710#							
L10115	024042R	002	5732#							
L10116	024156R	002	5777#							
L10117	024404R	002	5811	5852	5865#					
L10120	027536R	002	6248	6314	6357#					
L10121	027254R	002	6271#							
L10122	027534R	002	6354#							
L10123	030324R	002	6445	6488	6549	6596#				
L10124	030074R	002	6511#							
L10125	030304R	002	6585#							
L10126	031004R	002	6675	6695	6722	6742	6789#			
L10127	031422R	002	6877	6935#						
L10130	031712R	002	7024	7055#						
L10131	032420R	002	7110	7126	7136	7172	7246#			
L10132	032736R	002	7292	7307	7317	7339	7351	7382#		
L10133	033404R	002	7438	7453	7463	7486	7507	7516	7561#	
L10134	033762R	002	7603	7618	7628	7656	7671	7680	7710	7714#
L10135	034342R	002	7762	7777	7787	7810	7824	7833	7866	7873#
L10136	034670R	002	7957	7971	8005#					
L10137	035200R	002	8050	8065	8075	8096	8117	8126	8142#	
L10140	035510R	002	8188	8203	8213	8234	8255	8264	8280#	
L10141	036070R	002	8336	8351	8361	8384	8403	8412	8443#	
L10142	036450R	002	8498	8513	8523	8545	8564	8573	8604#	
L10143	037044R	002	8651	8666	8676	8707	8721	8730	8763#	
L10144	037606R	002	8828	8843	8853	8874	8892	8901	8987#	
L10145	040204R	002	9055	9070	9080	9110	9124	9133	9170	9179#
L10146	040606R	002	9224	9239	9249	9279	9293	9302	9340	9349#
L10147	041126R	002	9397	9412	9422	9443	9458	9467	9493#	
L10150	041516R	002	9562	9577	9587	9620	9634	9643	9677	9688#
L10151	042102R	002	9751	9766	9776	9806	9820	9829	9858	9871#
L10152	042466R	002	9914	9929	9939	9969	9983	9992	10020	10032#
L10153	043126R	002	10153	10168	10178	10220	10234	10243	10293#	
L10154	043456R	002	10376	10391	10401	10430	10453	10482#		
L10155	043776R	002	10531	10546	10556	10576	10592	10601	10628#	
L10156	044346R	002	10691	10706	10716	10742	10757	10766	10795	10807#
L10157	044666R	002	10858	10873	10883	10902	10924	10933	10960#	
L10160	045226R	002	11030	11045	11055	11081	11100	11109	11141#	
L10161	045616R	002	11198	11213	11223	11251	11270	11279	11322#	

```
L10162  046200R    002  11387   11402   11412   11454   11468   11477   11524#
L10163  046632R    002  11595   11610   11620   11653   11667   11676   11733   11744#
L10164  047236R    002  11810   11825   11835   11864   11878   11887   11941#
L10165  050326R    002  12046   12061   12071   12105   12119   12128   12182   12202   12254   12268   12277   12314#
L10166  051050R    002  12389   12404   12414   12438   12452   12461   12505   12519   12528   12543   12565#
L10167  051464R    002  12636   12651   12661   12690   12704   12713   12726   12757   12765#
L10170  052540R    002  12791   12797   12818   12831   12852   12865   12896   12909   12941   12953   13014#
L10171  053276R    002  13139   13153#
MAXBYT= 002756 G         1695#   9176    9346    9785    9948
MAXCNT  003027RG   002   2206#   9408    9418    9439    9454    9463    9476    9486
METER   000332RG   002   1830#   4278    4457*   4463    4530*   4651*   4679*   4878*   4989    4991*   5388*   5741*   5812*
                         6233*   6298*   6430*   6473*   6535*   6660*   6707*   6755*   6835*   6886*   6981*   7111*   7175*
                         7185    7293*   7340*   7356    7359    7439*   7495*   7604*   7659*   7763*   7812*   7913*   7959*
                         8051*   8098*   8189*   8236*   8337*   8391*   8499*   8552*   8652*   8709*   8829*   8880*   9056*
                         9112*   9225*   9281*   9398*   9446*   9563*   9622*   9752*   9808*   9915*   9971*   10154*  10222*
                        10377*  10432*  10532*  10577*  10692*  10745*  10859*  10912*  11031*  11088*  11199*  11258*  11388*
                        11456*  11596*  11655*  11811*  11866*  12047*  12107*  12184*  12256*  12390*  12440*  12507*  12637*
                        12692*  12802*  12836*  12880*  12925*
MICASZ= ****** GX        4586    6647
MICBSZ= ****** GX        4593
MICCSZ= ****** GX        4600
MICDSZ= ****** GX        4607
MICESZ= ****** GX        4614
MICFSZ= ****** GX        4621
MICGSZ= ****** GX        4625
MICMOD  000320RG   002   1823#   4571*   4575    4641    4656    4669    4684    4697
MICRO   000326RG   002   1828#   4571    4583    4590    4597    4604    4611    4618    6828    6832*   7101    7105*   7283
                         7287*   7429    7433*   7594    7598*   7753    7757*   7906    7910*   8041    8045*   8179    8183*
                         8327    8331*   8489    8493*   8642    8646*   8819    8823*   9046    9050*   9215    9219*   9388
                         9392*   9553    9557*   9742    9746*   9905    9909*   10144   10148*  10367   10371*  10522   10526*
                        10682   10686*  10849   10853*  11021   11025*  11189   11193*  11378   11382*  11586   11590*  11801
                        11805*  12037   12041*  12380   12384*  12627   12631*
MICROA= ****** GX        4587    6648
MICROB= ****** GX        4594
MICROC= ****** GX        4601
MICROD= ****** GX        4608
MICROE= ****** GX        4615
MICROF= ****** GX        4622
MICROG= ****** GX        4626
MINBYT= 000100 G         1696#   9088    9257    11740   12426   12670
MSG1    012716RG   002   2991#   5754    5825    6245    6311    6442    6485    6546    6672    6719    6767    6846    6993
                         7123    7304    7450    7615    7774    7924    8062    8200    8348    8510    8663    8840    9067
                         9236    9409    9574    9763    9926    10165   10388   10543   10703   10870   11042   11210   11399
                        11607   11822   12058   12401   12648   12815   12849   12893   12938
MSG10   013360RG   002   3142#   7348
MSG11   013402RG   002   3154#   7369
MSG12   013466RG   002   3179#   6902    7513    7677    7830    7977    8123    8261    8409    8570    8727    8898    9130
                         9299    9464    9640    9826    9989    10240   10598   10763   10930   11106   11276   11474   11673
                        11884   12125   12199   12274   12458   12525   12710
MSG13   013512RG   002   3192#   7694
MSG14   013560RG   002   3212#   7849
MSG15   013606RG   002   3226#   7994
MSG16   013636RG   002   3241#   6922    7536
MSG17   013730RG   002   3269#   8427    8588    8745
MSG18   014022RG   002   3297#   8919    8932    8941    8953    8962    8971
MSG19   014052RG   002   3311#   9153    9323
```

```
MSG2     012764RG   002    3008#   6351
MSG20    014146RG   002    3340#  10257
MSG21    014236RG   002    3367#  10273
MSG22    014326RG   002    3394#  12147   12219   12298
MSG23    014442RG   002    3429#  12472   12540
MSG24    014464RG   002    3441#  12754
MSG25    014614RG   002    3479#  10458
MSG26    015002RG   002    3527#  10776
MSG27    015046RG   002    3546#  11292
MSG28    015114RG   002    3566#  11300
MSG29    015162RG   002    3586#  11488
MSG3     013030RG   002    3027#   5398    5427    5451    5506    5531    5550
MSG30    015264RG   002    3614#  11690
MSG31    015332RG   002    3633#  11705
MSG32    015476RG   002    3674#  11716
MSG33    015574RG   002    3701#   7236
MSG34    015622RG   002    3714#   5849
MSG35    015672RG   002    3734#   9660
MSG36    015770RG   002    3761#   8107
MSG37    016012RG   002    3773#   8245
MSG38    016034RG   002    3785#   9477
MSG39    016056RG   002    3797#  11898
MSG4     013076RG   002    3044#   5482
MSG40    016100RG   002    3809#  11913
MSG41    016176RG   002    3836#  11924
MSG42    016250RG   002    3856#   9841   10003
MSG43    016420RG   002    3898#   7543
MSG44    016442RG   002    3910#   6895    7504    7668    7821    7968    8114    8252    8400    8561    8718    8389    9121
                           9290    9455    9631    9817    9980   10231   10450   10589   10754   10921   11097   11267   11465
                          11664   11875   12116   12193   12265   12449   12516   12701
MSG45    016604RG   002    3949#
MSG46    016666RG   002    3973#   6874    7169    7336    7483    7653    7807    7954    8093    8231    8381    8542    8704
                           8871    9107    9276    9440    9617    9803    9966   10217   10427   10573   10739   10899   11078
                          11248   11451   11650   11861   12102   12179   12251   12435   12502   12687
MSG5     013124RG   002    3057#   6580
MSG6     013174RG   002    3078#   6777
MSG7     013262RG   002    3104#   7021
MSG8     013310RG   002    3117#   7037
MSG9     013332RG   002    3129#   7197
MUCAST   003470RG   002    2259#  10869   10879   10898   10920   10929   10943   10953
NEXMEM   000666RG   002    1939#   4873    4944*   5028*   5031    5131*   5134    5277*   5280    5327*   5330
NIHLT  = 000006 G          1763#
NIUNI  = 000007 G          1764#
NOCLK    021610RG   002    4796    4849#
NOPERR   001706RG   002    2096#   5706    5728    5753    5770
NPRERR=  100000 G          1727#   3911    4496    4498
NXMERR=  040000 G          1728#   3919    4496    4498
ODDBYT   002765RG   002    2200#   9235    9245    9275    9289    9298    9322    9336
ONEFIL=  000001               1#   1391#   1417    1592#   1594    4713#   4723    4998    4999#  13124#  13126
OWN    = 100000 G          1767#
OSAPTS=  000000            1416#   1456
OSAU   = 000001            1416#   1427#   1488
OSBGNR=  000001            1416#   1427#   1482
OSBGNS=  000000            1416#   1448
OSDU   = 000001            1416#   1427#   1490
OSERRT=  000000            1416#   1498
```

```
O$GNSW= 000000            1416#   1452
O$POIN= 000001            1416#   1427#   1514
O$SETU= 000000            1416#   1442
PARERR= 010000 G          1730#   3935    4496    4498    7537    7544
PATERN  000520RG    002   1897#   5583    6398    7637    8689    11064   11234   11420
PAT1    000520RG    002   1898#   6291
PAT2    000522RG    002   1899#
PAT3    000524RG    002   1900#
PAT4    000526RG    002   1901#
PAT5    000530RG    002   1902#
PAT6    000532RG    002   1903#   6591
PCBB    000606RG    002   1927#   3405    3415    3588    3596    3603    3622    3662    3682    3690    3736    3749    3750
                          3824    3844    3858    3865    3879    3886    3887    4632*   4633*   4634*   4635    4706*   4707*
                          6230    6280*   6281*   6282*   6423*   6424*   6425*   6426    6533*   6653*   6654*   6655*   6656
                          6752*   6753*   6878    6880*   6881*   6882*   6883*   6906    6910    6915    6916    6917    7174*
                          7487    7489*   7490*   7491*   7492*   7522    7526    7529    7530    7531    7657*   7685    7687
                          7689    7792*   7840    7842    7867    7869*   7947*   7985    8385*   8387*   8388*   8389*   8419
                          8420    8421    8422    8546*   8548*   8549*   8550*   8580    8581    8582    8583    8685*   8687*
                          8688*   8691*   8737    8738    8739    8740    8875*   8878*   8912    8925    8934    8946    8955
                          8964    9088*   9097*   9099*   9141    9143    9144    9175*   9176    9257*   9266*   9268*   9310
                          9312    9313    9345*   9346    9608*   9609*   9652*   9653*   9654    9786*   9788*   9865    9867*
                          9949*   9951*   10026   10028*  10412*  10413*  10414*  10743*  11066*  11232*  11236*  11284   11426*
                          11439*  11482   11635*  11636*  11638*  11697*  11698   11700*  11710   11846*  11848*  11905*  11906
                          11908*  11918   12081   12093*  12094*  12095*  12135   12137   12139   12141   12170*  12171*  12172*
                          12207   12209   12211   12213   12242*  12243*  12244*  12286   12288   12290   12292   12426*  12427*
                          12466   12490*  12533   12562   12670*  12671*  12731   12799   12835*  12869   12870   12871   12879*
                          12913   12918*  12919*  12920*
PCEI  = 040000 G          1707#   4344    4347
PCEIB = 000100 G          1708#   4346
PCOMND  000316RG    002   1822#   2993    5749*   5820*   6240*   6306*   6437*   6480*   6541*   6667*   6714*   6762*   6841*
                          6988*   7118*   7299*   7445*   7610*   7769*   7919*   8057*   8195*   8343*   8505*   8658*   8835*
                          9062*   9231*   9404*   9569*   9758*   9921*   10160*  10383*  10538*  10698*  10865*  11037*  11205*
                          11394*  11602*  11817*  12053*  12396*  12643*  12810*  12844*  12888*  12933*
PCSR0   000336RG    002   1833#   4274    4309    4337    4340    4344    4347    4351    4354    4358    4361    4365    4368
                          4372    4375    4379    4382    4419    4495    4498*   4529*   4650*   4678*   4708*   4830*   4831
                          4833    4875*   4963    5030    5065    5082    5386*   5418    5443    5695    5740*   5813*   6232*
                          6299*   6429*   6472*   6534*   6659*   6706*   6754*   6884*   6979*   6980*   7045*   7176*   7178*
                          7219*   7220    7341*   7493*   7537    7544*   7658*   7811*   7958*   8097*   8235*   8390*   8551*
                          8708*   8879*   9111*   9280*   9444*   9621*   9807*   9970*   10221*  10431*  10579*  10744*  10907*
                          11087*  11257*  11455*  11654*  11865*  12106*  12183*  12255*  12439*  12506*  12691*  12801*  12837*
                          12881*  12926*
PCSR0C  000350RG    002   1839#
PCSR0U  000346RG    002   1837#   4308*   4339*   4346*   4353*   4360*   4367*   4374*   4381*   4831*   4832*
PCSR1   000340RG    002   1834#   3480    4418    4459    4833*   4834*   4835    5133    5168    5185    5202    5219    5236
                          5473    5498    5522    5541    5828    6768    6830    6904    7103    7285    7431    7520    7596
                          7755    7908    8043    8181    8329    8417    8491    8578    8644    8735    8821    9048    9217
                          9390    9471    9555    9744    9835    9907    9997    10146   10248   10369   10524   10606   10684
                          10770   10851   10938   11023   11113   11191   11380   11588   11684   11803   11892   12039   12382
                          12629
PCSR1C  000352RG    002   1840#
PCSR2   000342RG    002   1835#   4417    4635*   4835*   4836*   4837    5279    5588*   5589    6230*   6426*   6656*   6878*
                          7487*   10187*  12081*  12799*
PCSR2C  000354RG    002   1841#
PCSR3   000344RG    002   1836#   4416    4636*   4837*   4838*   5329    5630*   5631    5646*   5647    6231*   6427*   6657*
                          6879*   7488*   10188*  12082*  12800*
PCSR3C  000356RG    002   1842#   4415
```

```
PCTO   = 000200 G           1744#
PFNOP1= 000000 G            1775#
PFNOP2= 000003 G            1778#
PNOP   = 000006 G           1735#   5740    6980
PNT    = 001000 G           1664#
POLYH  = 166670 G           1690#   4052
POLYL  = 101440 G           1691#   4053
POLY16= 120001 G            1692#   4095
PRI    = 002000 G           1665#
PRILD  = 000001 G           1759#
PR100  = 000000 G           1653#
PR101  = 000040 G           1652#
PR102  = 000100 G           1651#
PR103  = 000140 G           1650#
PR104  = 000200 G           1649#   7181
PR105  = 000240 G           1648#   4233    4829
PR106  = 000300 G           1647#
PR107  = 000340 G           1646#   4247    5016    5119    5271    5319
PRNPCR   017764R     002    2938    2972    2985    3002    3038    3051    3111    4413#   4548    4887
PRTPAR   004061RG    002    2306#   12814   12827   12848   12861   12892   12905   12937   12949
PSTATE= 000007 G            1757#   5474    6769
PWHEN    000314RG    002    1821#   2994    3029    4536*   4539    5378*   5493*   5748*   5819*   6239*   6305*   6436*   6479*
                            6540*   6666*   6713*   6761*   6840*   6987*   7117*   7298*   7444*   7609*   7768*   7918*   8056*
                            8194*   8342*   8504*   8657*   8834*   9061*   9230*   9403*   9568*   9757*   9920*   10159*  10382*
                            10537*  10697*  10864*  11036*  11204*  11393*  11601*  11816*  12052*  12395*  12642*  12809*  12843*
                            12887*  12932*
RACC   = 000013 G           1786#
RACERR   001570RG    002    2082#   5036    5070    5087    5139    5173    5190    5207    5224    5241    5285    5335
RACMG1   012500RG    002    2915#   5037    5140    5286    5336
RACMG2   012526RG    002    2928#   5071    5088    5174    5191    5208    5225    5242    7209
RACMG3   012570RG    002    2944#   5598    5638    5654
RACMG4   012642RG    002    2965#   5707
RACMG7   012670RG    002    2978#   5729    5771    5862    6265    6331    6462    6505    6563    6692    6739    6854    6929
                            7052    7133    7314    7376    7460    7555    7625    7707    7784    7863    7932    8072    8136
                            8210    8274    8358    8437    8520    8598    8673    8755    8850    8981    9077    9167    9246
                            9337    9419    9487    9584    9674    9773    9855    9936    10017   10175   10287   10398   10469
                            10553   10622   10713   10792   10880   10954   11052   11129   11220   11310   11409   11498   11617
                            11730   11832   11938   12068   12158   12230   12308   12411   12482   12553   12658   12723   12828
                            12862   12906   12950
RACPS  = 000017 G           1790#
RBRRUN   003650RG    002    2279#   11606   11616   11649   11663   11672   11689   11704   11715   11729
RC     = 000012 G           1785#
RCEI   = 002000 G           1715#   4372    4375
RCEIB  = 000004 G           1716#   4374
RCVDON   002477RG    002    2164#   8199    8209    8230    8244    8251    8260    8273
RDPA   = 000002 G           1777#   12835
READY  = 000002 G           1760#   5475
RESET  = 000000 G           1758#
RETLOG   004031RG    002    2301#   12647   12657   12686   12700   12709   12722   12753
REUNA    020166RG    002    4529#   4572    6786    6930    8000    12793
RLNKAD   003114RG    002    2216#   9762    9772    9802    9816    9825    9840    9854
RLSA   = 000024 G           1795#
RMAL   = 000006 G           1781#
RMTC   = 000010 G           1746#
ROMADR= 040000 G            1685#   1686    6283
ROMDMP   001747RG    002    2102#   6244    6264    6310    6330    6350
```

```
ROMSIZ= 040000 G          1681#   1686    6295    6342
RPA   = 000004 G          1779#
RPS   = 000016 G          1789#   12879
RREV    052550RG    002   12913*  12914*  12984   13021#
RRF   = 000010 G          1783#
RRWER   001646RG    002   2090#   5597    5637    5653
RSET  = 000040 G          1722#   4529    4875    5386
RSETER  001624RG    002   2087#   5397    5426    5450    5481    5505    5530    5549
RSIDP = 000022 G          1793#
RTM   = 000014 G          1787#
RUN   = 000003 G          1761#
RXI   = 020000 G          1709#   4351    4354
RXIB  = 000040 G          1710#   4353
SECOND= 000077 G          1672#   4457    4530    4651    4679    4878    5388    5741    5812    6233    6298    6430    6473
                          6535    6660    6707    6755    6835    6886    6981    7111    7175    7293    7439    7495    7604
                          7659    7763    7812    7913    7959    8051    8098    8189    8236    8337    8391    8499    8552
                          8652    8709    8829    8880    9056    9112    9225    9281    9398    9446    9563    9622    9752
                          9808    9915    9971    10154   10222   10377   10432   10532   10577   10692   10745   10859   10912
                          11031   11088   11199   11258   11388   11456   11596   11655   11811   11866   12047   12107   12184
                          12256   12390   12440   12507   12637   12692   12802   12836   12880   12925
SERI  = 100000 G          1705#   4337    4340
SERIB = 000200 G          1706#   4339
SET   = 000001 G          1673#
SFT   = 037400 G          1753#   5829    5832    6769
SFTB0 = 000400 G          1747#   6770
SFTB1 = 001000 G          1748#
SFTB2 = 002000 G          1749#   6770
SFTB3 = 004000 G          1750#
SFTB4 = 010000 G          1751#   6770
SFTB5 = 020000 G          1752#
SIZ1K = 004000 G          1675#   1676    6290    6340    9597    9682    9684    9788    9865    9867    9951    10026   10028
                          10199
SIZ2K = 010000 G          1676#   1677
SIZ4K = 020000 G          1677#   1678    1679    1680
SIZ8K = 040000 G          1678#   1681    1682
SLFT  = 000003 G          1736#   5813
SLFTST  001726RG    002   2099#   5807    5824    5848    5861
SMSG00  024610RG    002   5874    5939#
SMSG01  024632RG    002   5875    5942#
SMSG02  024652RG    002   5876    5945#
SMSG03  024656RG    002   5877    5946#
SMSG04  024706RG    002   5878    5951#
SMSG05  024742RG    002   5879    5956#
SMSG06  024766RG    002   5880    5960#
SMSG07  025030RG    002   5881    5966#
SMSG10  025073RG    002   5882    5972#
SMSG11  025140RG    002   5883    5979#
SMSG12  025160RG    002   5884    5982#
SMSG13  025166RG    002   5885    5983#
SMSG14  025213RG    002   5886    5987#
SMSG15  025214RG    002   5887    5988#
SMSG16  025215RG    002   5888    5989#
SMSG17  025216RG    002   5889    5990#
SMSG20  025217RG    002   5890    5991#
SMSG21  025233RG    002   5891    5994#
SMSG22  025234RG    002   5892    5995#
```

```
SMSG23  025235RG   002   5893    5996#
SMSG24  025236RG   002   5894    5997#
SMSG25  025237RG   002   5895    5998#
SMSG26  025240RG   002   5896    5999#
SMSG27  025241RG   002   5897    6000#
SMSG30  025242RG   002   5898    6001#
SMSG31  025304RG   002   5899    6007#
SMSG32  025347RG   002   5900    6013#
SMSG33  025413RG   002   5901    6020#
SMSG34  025450RG   002   5902    6025#
SMSG35  025512RG   002   5903    6031#
SMSG36  025546RG   002   5904    6036#
SMSG37  025610RG   002   5905    6042#
SMSG40  025644RG   002   5906    6047#
SMSG41  025721RG   002   5907    6055#
SMSG42  025773RG   002   5908    6063#
SMSG43  026046RG   002   5909    6071#
SMSG44  026112RG   002   5910    6077#
SMSG45  026163RG   002   5911    6084#
SMSG46  026226RG   002   5912    6090#
SMSG47  026227RG   002   5913    6091#
SMSG50  026230RG   002   5914    6092#
SMSG51  026307RG   002   5915    6100#
SMSG52  026363RG   002   5916    6108#
SMSG53  026440RG   002   5917    6116#
SMSG54  026506RG   002   5918    6123#
SMSG55  026561RG   002   5919    6131#
SMSG56  026626RG   002   5920    6138#
SMSG57  026627RG   002   5921    6139#
SMSG60  026630RG   002   5922    6140#
SMSG61  026644RG   002   5923    6142#
SMSG62  026670RG   002   5924    6146#
SMSG63  026714RG   002   5925    6150#
SMSG64  026734RG   002   5926    6153#
SMSG65  026740RG   002   5927    6154#
SMSG66  026752RG   002   5928    6156#
SMSG67  026764RG   002   5929    6158#
SMSG70  027000RG   002   5930    6160#
SMSG71  027012RG   002   5931    6162#
SMSG72  027036RG   002   5932    6166#
SMSG73  027060RG   002   5933    6169#
SMSG74  027061RG   002   5934    6170#
SMSG75  027062RG   002   5935    6171#
SMSG76  027063RG   002   5936    6172#
SMSG77  027064RG   002   5937    6173#
SPAT1   000534RG   002   1905#   10410
SPAT2   000542RG   002   1908#
SPAT3   000550RG   002   1911#
SPAT4   000556RG   002   1914#
SPAT5   000564RG   002   1917#
SPAT6   000572RG   002   1920#
SPAT7   000600RG   002   1923#
STAMUX  002663RG   002   2187#    8839    8849    8870    8888    8897    8918    8931    8940    8952    8961    8970    8980
STAPAT  003305RG   002   2238#   10387   10397   10426   10449   10457   10468
STAPOS  003417RG   002   2252#   10702   10712   10738   10753   10762   10775   10791
STAREJ  003351RG   002   2245#   10542   10552   10572   10588   10597   10611   10621
```

```
STMSG    024406RG   002   3723    5842*   5844*   5870#
STP    = 001000 G         1769#
STTBL    024410RG   002   5839    5874#
SVCGBL= 000001            1416#   1429    1438    1440    1442    1444    1446    1448    1450    1452    1454    1456    1458
                          1460    1462    1464    1466    1468    1470    1472    1475    1478    1480    1482    1484    1486
                          1488    1490    1492    1494    1496    1498    1500    1502    1504    1506    1508    1510    1512
                          1524    1583    1584    1962    1970    2915    2928    2944    2965    2978    2991    3008    3027
                          3044    3057    3078    3104    3117    3129    3142    3154    3179    3192    3212    3226    3241
                          3269    3297    3311    3340    3367    3394    3429    3441    3479    3527    3546    3566    3586
                          3614    3633    3674    3701    3714    3734    3761    3773    3785    3797    3809    3836    3856
                          3898    3910    3949    3973    4716    4733    4749    4860    4872    4909    4920    4943    4962
                          4988    13140
SVCINS= 000001            1416#   1430    1431    1432    1433    1434    1435    1436    1437    1439    1441    1443    1445
                          1447    1449    1451    1453    1455    1457    1459    1461    1463    1465    1467    1469    1471
                          1473    1474    1476    1477    1479    1481    1483    1485    1487    1489    1491    1493    1495
                          1497    1499    1501    1503    1505    1507    1509    1511    1513    1523    1525    1526    1527
                          1528    1529    1530    1531    1532    1533    1534    1535    1536    1537    1538    1539    1540
                          1541    1542    1543    1544    1545    1546    1547    1548    1549    1550    1551    1552    1553
                          1554    1555    1556    1557    1558    1559    1560    1561    1562    1563    1564    1565    1566
                          1567    1568    1569    1570    1582    1963    1964    1971    1975    2917    2918    2919    2920
                          2921    2922    2925    2930    2931    2932    2933    2934    2935    2936    2937    2941    2946
                          2947    2948    2949    2950    2951    2953    2954    2955    2956    2957    2958    2959    2962
                          2967    2968    2969    2970    2971    2975    2980    2981    2982    2983    2984    2988    2993
                          2994    2995    2996    2997    2998    2999    3000    3001    3005    3010    3011    3012    3013
                          3014    3016    3017    3018    3019    3020    3021    3024    3029    3030    3031    3032    3033
                          3034    3035    3036    3037    3041    3046    3047    3048    3049    3050    3054    3059    3060
                          3061    3062    3063    3065    3066    3067    3068    3069    3070    3071    3072    3075    3080
                          3081    3082    3083    3084    3086    3087    3088    3089    3090    3091    3093    3094    3095
                          3096    3097    3098    3101    3106    3107    3108    3109    3110    3114    3119    3120    3121
                          3122    3123    3126    3131    3132    3133    3134    3135    3136    3139    3144    3145    3146
                          3147    3148    3151    3156    3157    3158    3159    3160    3162    3163    3164    3165    3166
                          3168    3169    3170    3171    3172    3173    3176    3181    3182    3183    3184    3185    3186
                          3189    3194    3195    3196    3197    3198    3199    3201    3202    3203    3204    3205    3206
                          3209    3214    3215    3216    3217    3218    3219    3220    3223    3228    3229    3230    3231
                          3232    3233    3234    3235    3238    3243    3244    3245    3246    3247    3248    3250    3251
                          3252    3253    3254    3255    3257    3258    3259    3260    3261    3262    3263    3266    3271
                          3272    3273    3274    3275    3276    3278    3279    3280    3281    3282    3283    3285    3286
                          3287    3288    3289    3290    3291    3294    3299    3300    3301    3302    3303    3304    3305
                          3308    3313    3314    3315    3316    3317    3318    3320    3321    3322    3323    3324    3325
                          3326    3328    3329    3330    3331    3332    3333    3334    3337    3342    3343    3344    3345
                          3346    3348    3349    3350    3351    3352    3353    3355    3356    3357    3358    3359    3360
                          3361    3364    3369    3370    3371    3372    3373    3375    3376    3377    3378    3379    3380
                          3382    3383    3384    3385    3386    3387    3388    3391    3397    3398    3399    3400    3401
                          3402    3405    3406    3407    3408    3409    3410    3411    3412    3415    3416    3417    3418
                          3419    3420    3421    3422    3426    3431    3432    3433    3434    3435    3438    3443    3444
                          3445    3446    3447    3449    3450    3451    3452    3453    3458    3459    3460    3461    3462
                          3463    3465    3466    3467    3468    3469    3470    3476    3483    3484    3485    3486    3487
                          3490    3491    3492    3493    3494    3496    3497    3498    3499    3500    3502    3503    3504
                          3505    3506    3507    3509    3510    3511    3512    3513    3514    3516    3517    3518    3519
                          3520    3521    3524    3529    3530    3531    3532    3533    3535    3536    3537    3538    3539
                          3540    3543    3548    3549    3550    3551    3552    3554    3555    3556    3557    3558    3559
                          3560    3563    3568    3569    3570    3571    3572    3574    3575    3576    3577    3578    3579
                          3580    3583    3588    3589    3590    3591    3592    3593    3594    3596    3597    3598    3599
                          3600    3601    3603    3604    3605    3606    3607    3608    3611    3616    3617    3618    3619
                          3620    3622    3623    3624    3625    3626    3627    3630    3635    3636    3637    3638    3639
                          3641    3642    3643    3644    3645    3646    3648    3649    3650    3651    3652    3653    3655
```

3656	3657	3658	3659	3660	3662	3663	3664	3665	3666	3667	3668	3671
3676	3677	3678	3679	3680	3682	3683	3684	3685	3686	3687	3688	3690
3691	3692	3693	3694	3695	3698	3703	3704	3705	3706	3707	3708	3711
3716	3717	3718	3719	3720	3721	3723	3724	3725	3726	3727	3728	3731
3736	3737	3738	3739	3740	3741	3743	3744	3745	3746	3747	3749	3750
3751	3752	3753	3754	3755	3758	3763	3764	3765	3766	3767	3770	3775
3776	3777	3778	3779	3782	3787	3788	3789	3790	3791	3794	3799	3800
3801	3802	3803	3806	3811	3812	3813	3814	3815	3817	3818	3819	3820
3821	3822	3824	3825	3826	3827	3828	3829	3830	3833	3838	3839	3840
3841	3842	3844	3845	3846	3847	3848	3849	3850	3853	3858	3859	3860
3861	3862	3863	3865	3866	3867	3868	3869	3870	3872	3873	3874	3875
3876	3877	3879	3880	3881	3882	3883	3884	3886	3887	3888	3889	3890
3891	3892	3895	3900	3901	3902	3903	3904	3907	3914	3915	3916	3917
3918	3922	3923	3924	3925	3926	3930	3931	3932	3933	3934	3938	3939
3940	3941	3942	3946	3951	3952	3953	3954	3955	3957	3958	3959	3960
3961	3963	3964	3965	3966	3967	3970	3975	3976	3977	3978	3979	3982
4233	4234	4247	4248	4277	4386	4387	4388	4389	4390	4391	4422	4423
4424	4425	4426	4427	4428	4462	4539	4540	4541	4542	4543	4544	4545
4546	4547	4554	4555	4556	4557	4558	4575	4576	4577	4578	4579	4580
4641	4642	4643	4644	4645	4646	4656	4657	4658	4659	4660	4661	4669
4670	4671	4672	4673	4674	4684	4685	4686	4687	4688	4689	4697	4698
4699	4700	4701	4702	4718	4719	4722	4754	4755	4757	4759	4760	4763
4774	4775	4777	4779	4780	4782	4785	4786	4790	4791	4792	4794	4796
4797	4798	4799	4800	4809	4810	4811	4812	4813	4814	4822	4823	4824
4826	4842	4846	4863	4882	4883	4884	4885	4886	4891	4892	4905	4911
4912	4915	4922	4923	4926	4947	4967	4995	5016	5017	5018	5019	5020
5021	5024	5034	5035	5036	5037	5039	5041	5042	5044	5045	5047	5051
5054	5059	5068	5069	5070	5071	5075	5077	5085	5086	5087	5088	5092
5095	5097	5098	5101	5119	5120	5121	5122	5123	5124	5127	5137	5138
5139	5140	5142	5144	5145	5147	5148	5150	5154	5157	5162	5171	5172
5173	5174	5178	5180	5188	5189	5190	5191	5195	5197	5205	5206	5207
5208	5212	5214	5222	5223	5224	5225	5229	5231	5239	5240	5241	5242
5246	5249	5251	5252	5255	5271	5272	5273	5274	5275	5276	5283	5284
5285	5286	5288	5290	5291	5293	5294	5296	5298	5299	5303	5319	5320
5321	5322	5323	5324	5326	5333	5334	5335	5336	5338	5340	5341	5343
5344	5346	5350	5352	5353	5356	5382	5395	5396	5397	5398	5400	5401
5405	5409	5414	5424	5425	5426	5427	5431	5442	5448	5449	5450	5451
5455	5468	5479	5480	5481	5482	5486	5497	5503	5504	5505	5506	5510
5518	5528	5529	5530	5531	5535	5537	5547	5548	5549	5550	5554	5557
5560	5587	5595	5596	5597	5598	5602	5607	5629	5635	5636	5637	5638
5642	5645	5651	5652	5653	5654	5658	5661	5691	5704	5705	5706	5707
5711	5714	5726	5727	5728	5729	5733	5736	5751	5752	5753	5754	5756
5757	5760	5768	5769	5770	5771	5775	5778	5781	5805	5806	5807	5808
5810	5811	5822	5823	5824	5825	5846	5847	5848	5849	5851	5852	5859
5860	5861	5862	5866	6224	6229	6242	6243	6244	6245	6247	6248	6252
6254	6262	6263	6264	6265	6269	6272	6275	6288	6308	6309	6310	6311
6313	6314	6318	6320	6328	6329	6330	6331	6335	6348	6349	6350	6351
6355	6358	6401	6406	6439	6440	6441	6442	6444	6445	6449	6451	6459
6460	6461	6462	6466	6468	6482	6483	6484	6485	6487	6488	6492	6494
6502	6503	6504	6505	6509	6512	6515	6520	6543	6544	6545	6546	6548
6549	6553	6555	6560	6561	6562	6563	6567	6577	6578	6579	6580	6586
6597	6641	6669	6670	6671	6672	6674	6675	6679	6681	6689	6690	6691
6692	6694	6695	6699	6701	6716	6717	6718	6719	6721	6722	6726	6728
6736	6737	6738	6739	6741	6742	6746	6751	6764	6765	6766	6767	6774
6775	6776	6777	6781	6790	6827	6843	6844	6845	6846	6851	6852	6853
6854	6859	6867	6871	6872	6873	6874	6876	6877	6892	6893	6894	6895

6899	6900	6901	6902	6919	6920	6921	6922	6926	6927	6928	6929	6933
6936	6970	6972	6973	6974	6975	6976	6977	6990	6991	6992	6993	6997
7002	7008	7009	7018	7019	7020	7021	7023	7024	7034	7035	7036	7037
7041	7043	7044	7049	7050	7051	7052	7056	7100	7109	7110	7120	7121
7122	7123	7125	7126	7130	7131	7132	7133	7135	7136	7138	7139	7140
7141	7142	7143	7146	7162	7166	7167	7168	7169	7171	7172	7181	7182
7194	7195	7196	7197	7206	7207	7208	7209	7213	7218	7233	7234	7235
7236	7240	7247	7282	7291	7292	7301	7302	7303	7304	7306	7307	7311
7312	7313	7314	7316	7317	7321	7329	7333	7334	7335	7336	7338	7339
7345	7346	7347	7348	7350	7351	7366	7367	7368	7369	7373	7374	7375
7376	7380	7383	7428	7437	7438	7447	7448	7449	7450	7452	7453	7457
7458	7459	7460	7462	7463	7467	7476	7480	7481	7482	7483	7485	7486
7501	7502	7503	7504	7506	7507	7510	7511	7512	7513	7515	7516	7533
7534	7535	7536	7540	7541	7542	7543	7552	7553	7554	7555	7559	7562
7593	7602	7603	7612	7613	7614	7615	7617	7618	7622	7623	7624	7625
7627	7628	7632	7646	7650	7651	7652	7653	7655	7656	7665	7666	7667
7668	7670	7671	7674	7675	7676	7677	7679	7680	7691	7692	7693	7694
7697	7704	7705	7706	7707	7709	7710	7715	7752	7761	7762	7771	7772
7773	7774	7776	7777	7781	7782	7783	7784	7786	7787	7791	7800	7804
7805	7806	7807	7809	7810	7818	7819	7820	7821	7823	7824	7827	7828
7829	7830	7832	7833	7846	7847	7848	7849	7853	7860	7861	7862	7863
7865	7866	7874	7905	7921	7922	7923	7924	7929	7930	7931	7932	7937
7946	7951	7952	7953	7954	7956	7957	7965	7966	7967	7968	7970	7971
7974	7975	7976	7977	7991	7992	7993	7994	8003	8006	8040	8049	8050
8059	8060	8061	8062	8064	8065	8069	8070	8071	8072	8074	8075	8079
8086	8090	8091	8092	8093	8095	8096	8104	8105	8106	8107	8111	8112
8113	8114	8116	8117	8120	8121	8122	8123	8125	8126	8133	8134	8135
8136	8140	8143	8178	8187	8188	8197	8198	8199	8200	8202	8203	8207
8208	8209	8210	8212	8213	8217	8224	8228	8229	8230	8231	8233	8234
8242	8243	8244	8245	8249	8250	8251	8252	8254	8255	8258	8259	8260
8261	8263	8264	8271	8272	8273	8274	8278	8281	8326	8335	8336	8345
8346	8347	8348	8350	8351	8355	8356	8357	8358	8360	8361	8365	8374
8378	8379	8380	8381	8383	8384	8397	8398	8399	8400	8402	8403	8406
8407	8408	8409	8411	8412	8424	8425	8426	8427	8434	8435	8436	8437
8441	8444	8488	8497	8498	8507	8508	8509	8510	8512	8513	8517	8518
8519	8520	8522	8523	8527	8535	8539	8540	8541	8542	8544	8545	8558
8559	8560	8561	8563	8564	8567	8568	8569	8570	8572	8573	8585	8586
8587	8588	8595	8596	8597	8598	8602	8605	8641	8650	8651	8660	8661
8662	8663	8665	8666	8670	8671	8672	8673	8675	8676	8680	8693	8701
8702	8703	8704	8706	8707	8715	8716	8717	8718	8720	8721	8724	8725
8726	8727	8729	8730	8742	8743	8744	8745	8752	8753	8754	8755	8759
8764	8818	8827	8828	8837	8838	8839	8840	8842	8843	8847	8848	8849
8850	8852	8853	8857	8864	8868	8869	8870	8871	8873	8874	8886	8887
8888	8889	8891	8892	8895	8896	8897	8898	8900	8901	8916	8917	8918
8919	8929	8930	8931	8932	8938	8939	8940	8941	8950	8951	8952	8953
8959	8960	8961	8962	8968	8969	8970	8971	8978	8979	8980	8981	8985
8988	9045	9054	9055	9064	9065	9066	9067	9069	9070	9074	9075	9076
9077	9079	9080	9084	9096	9104	9105	9106	9107	9109	9110	9118	9119
9120	9121	9123	9124	9127	9128	9129	9130	9132	9133	9150	9151	9152
9153	9157	9164	9165	9166	9167	9169	9170	9180	9214	9223	9224	9233
9234	9235	9236	9238	9239	9243	9244	9245	9246	9248	9249	9253	9265
9273	9274	9275	9276	9278	9279	9287	9288	9289	9290	9292	9293	9296
9297	9298	9299	9301	9302	9320	9321	9322	9323	9327	9334	9335	9336
9337	9339	9340	9350	9387	9396	9397	9406	9407	9408	9409	9411	9412
9416	9417	9418	9419	9421	9422	9426	9433	9437	9438	9439	9440	9442
9443	9452	9453	9454	9455	9457	9458	9461	9462	9463	9464	9466	9467

9474	9475	9476	9477	9484	9485	9486	9487	9491	9494	9552	9561	9562
9571	9572	9573	9574	9576	9577	9581	9582	9583	9584	9586	9587	9591
9607	9614	9615	9616	9617	9619	9620	9628	9629	9630	9631	9633	9634
9637	9638	9639	9640	9642	9643	9657	9658	9659	9660	9664	9671	9672
9673	9674	9676	9677	9689	9741	9750	9751	9760	9761	9762	9763	9765
9766	9770	9771	9772	9773	9775	9776	9780	9792	9800	9801	9802	9803
9805	9806	9814	9815	9816	9817	9819	9820	9823	9824	9825	9826	9828
9829	9838	9839	9840	9841	9845	9852	9853	9854	9855	9857	9858	9872
9904	9913	9914	9923	9924	9925	9926	9928	9929	9933	9934	9935	9936
9938	9939	9943	9959	9963	9964	9965	9966	9968	9969	9977	9978	9979
9980	9982	9983	9986	9987	9988	9989	9991	9992	10000	10001	10002	10003
10007	10014	10015	10016	10017	10019	10020	10033	10143	10152	10153	10162	10163
10164	10165	10167	10168	10172	10173	10174	10175	10177	10178	10182	10205	10214
10215	10216	10217	10219	10220	10228	10229	10230	10231	10233	10234	10237	10238
10239	10240	10242	10243	10254	10255	10256	10257	10270	10271	10272	10273	10284
10285	10286	10287	10291	10294	10366	10375	10376	10385	10386	10387	10388	10390
10391	10395	10396	10397	10398	10400	10401	10405	10416	10424	10425	10426	10427
10429	10430	10447	10448	10449	10450	10452	10453	10455	10456	10457	10458	10466
10467	10468	10469	10473	10483	10521	10530	10531	10540	10541	10542	10543	10545
10546	10550	10551	10552	10553	10555	10556	10560	10566	10570	10571	10572	10573
10575	10576	10586	10587	10588	10589	10591	10592	10595	10596	10597	10598	10600
10601	10609	10610	10611	10612	10619	10620	10621	10622	10626	10629	10681	10690
10691	10700	10701	10702	10703	10705	10706	10710	10711	10712	10713	10715	10716
10720	10732	10736	10737	10738	10739	10741	10742	10751	10752	10753	10754	10756
10757	10760	10761	10762	10763	10765	10766	10773	10774	10775	10776	10780	10785
10789	10790	10791	10792	10794	10795	10799	10808	10848	10857	10858	10867	10868
10869	10870	10872	10873	10877	10878	10879	10880	10882	10883	10887	10892	10896
10897	10898	10899	10901	10902	10918	10919	10920	10921	10923	10924	10927	10928
10929	10930	10932	10933	10941	10942	10943	10944	10951	10952	10953	10954	10958
10961	11020	11029	11030	11039	11040	11041	11042	11044	11045	11049	11050	11051
11052	11054	11055	11059	11068	11075	11076	11077	11078	11080	11081	11094	11095
11096	11097	11099	11100	11103	11104	11105	11106	11108	11109	11116	11117	11118
11119	11126	11127	11128	11129	11134	11142	11188	11197	11198	11207	11208	11209
11210	11212	11213	11217	11218	11219	11220	11222	11223	11227	11241	11245	11246
11247	11248	11250	11251	11264	11265	11266	11267	11269	11270	11273	11274	11275
11276	11278	11279	11289	11290	11291	11292	11297	11298	11299	11300	11307	11308
11309	11310	11314	11323	11377	11386	11387	11396	11397	11398	11399	11401	11402
11406	11407	11408	11409	11411	11412	11416	11428	11438	11448	11449	11450	11451
11453	11454	11462	11463	11464	11465	11467	11468	11471	11472	11473	11474	11476
11477	11485	11486	11487	11488	11495	11496	11497	11498	11502	11515	11525	11585
11594	11595	11604	11605	11606	11607	11609	11610	11614	11615	11616	11617	11619
11620	11624	11634	11647	11648	11649	11650	11652	11653	11661	11662	11663	11664
11666	11667	11670	11671	11672	11673	11675	11676	11687	11688	11689	11690	11702
11703	11704	11705	11713	11714	11715	11716	11720	11727	11728	11729	11730	11732
11733	11745	11800	11809	11810	11819	11820	11821	11822	11824	11825	11829	11830
11831	11832	11834	11835	11839	11845	11858	11859	11860	11861	11863	11864	11872
11873	11874	11875	11877	11878	11881	11882	11883	11884	11886	11887	11895	11896
11897	11898	11910	11911	11912	11913	11921	11922	11923	11924	11928	11935	11936
11937	11938	11942	12036	12045	12046	12055	12056	12057	12058	12060	12061	12065
12066	12067	12068	12070	12071	12075	12091	12099	12100	12101	12102	12104	12105
12113	12114	12115	12116	12118	12119	12122	12123	12124	12125	12127	12128	12144
12145	12146	12147	12155	12156	12157	12158	12162	12167	12176	12177	12178	12179
12181	12182	12190	12191	12192	12193	12196	12197	12198	12199	12201	12202	12216
12217	12218	12219	12227	12228	12229	12230	12236	12241	12248	12249	12250	12251
12253	12254	12262	12263	12264	12265	12267	12268	12271	12272	12273	12274	12276
12277	12295	12296	12297	12298	12305	12306	12307	12308	12312	12315	12379	12388

```
                                12389   12398   12399   12400   12401   12403   12404   12408   12409   12410   12411   12413   12414
                                12418   12425   12432   12433   12434   12435   12437   12438   12446   12447   12448   12449   12451
                                12452   12455   12456   12457   12458   12460   12461   12469   12470   12471   12472   12479   12480
                                12481   12482   12486   12492   12499   12500   12501   12502   12504   12505   12513   12514   12515
                                12516   12518   12519   12522   12523   12524   12525   12527   12528   12537   12538   12539   12540
                                12542   12543   12550   12551   12552   12553   12557   12566   12626   12635   12636   12645   12646
                                12647   12648   12650   12651   12655   12656   12657   12658   12660   12661   12665   12680   12684
                                12685   12686   12687   12689   12690   12698   12699   12700   12701   12703   12704   12707   12708
                                12709   12710   12712   12713   12720   12721   12722   12723   12725   12726   12751   12752   12753
                                12754   12756   12757   12763   12766   12790   12791   12796   12797   12812   12813   12814   12815
                                12817   12818   12825   12826   12827   12828   12830   12831   12846   12847   12848   12849   12851
                                12852   12859   12860   12861   12862   12864   12865   12890   12891   12892   12893   12895   12896
                                12903   12904   12905   12906   12908   12909   12935   12936   12937   12938   12940   12941   12947
                                12948   12949   12950   12952   12953   12977   12978   12979   12980   12981   12982   12984   12985
                                12986   12987   12988   12989   12991   12992   12993   12994   12995   12996   12998   12999   13000
                                13001   13002   13003   13005   13006   13007   13008   13009   13010   13015   13139   13142   13143
                                13144   13145   13147   13148   13149   13150   13152
SVCSUB= 000001                   1416#   5023    5053    5126    5156    5381    5408    5690    5713    5735    6223    6274    6400
                                 6514
SVCTAG= 000001                   1416#   1590    2924    2940    2961    2974    2987    3004    3023    3040    3053    3074    3100
                                 3113    3125    3138    3150    3175    3188    3208    3222    3237    3265    3293    3307    3336
                                 3363    3390    3425    3437    3475    3523    3542    3562    3582    3610    3629    3670    3697
                                 3710    3730    3757    3769    3781    3793    3805    3832    3852    3894    3906    3945    3969
                                 3981    4721    4845    4862    4904    4914    4925    4946    4966    4994    5050    5074    5091
                                 5094    5100    5153    5177    5194    5211    5228    5245    5248    5254    5302    5349    5355
                                 5404    5430    5454    5485    5509    5534    5553    5556    5559    5601    5606    5641    5657
                                 5660    5710    5732    5774    5777    5780    5865    6251    6268    6271    6317    6334    6354
                                 6357    6448    6465    6491    6508    6511    6552    6566    6585    6596    6678    6698    6725
                                 6745    6780    6789    6858    6932    6935    6996    7040    7055    7145    7212    7239    7246
                                 7320    7379    7382    7466    7558    7561    7631    7696    7714    7790    7852    7873    7936
                                 8002    8005    8078    8139    8142    8216    8277    8280    8364    8440    8443    8526    8601
                                 8604    8679    8758    8763    8856    8984    8987    9083    9156    9179    9252    9326    9349
                                 9425    9490    9493    9590    9663    9688    9779    9844    9871    9942   10006   10032   10181
                                10290   10293   10404   10472   10482   10559   10625   10628   10719   10779   10798   10807   10886
                                10957   10960   11058   11133   11141   11226   11313   11322   11415   11501   11514   11524   11623
                                11719   11744   11838   11927   11941   12074   12161   12235   12311   12314   12417   12485   12556
                                12565   12664   12762   12765   13014   13153
SVCTST= 000001                   1416#   5014    5117    5269    5317    5376    5581    5625    5688    5801    6221    6397    6636
                                 6820    6964    7093    7275    7421    7586    7745    7898    8033    8171    8319    8481    8634
                                 8811    9038    9207    9380    9545    9734    9897   10136   10359   10514   10674   10841   11013
                                11181   11370   11578   11793   12029   12372   12619   12786
SWADDR   000334RG     002        1831#  12924
SWHDR    052706RG     002       12991   13077#
SWPACK   052552RG     002       12922   12957   12965   13023#
SSLSYM= 010000                   1416#   1591#   2925#   2941#   2962#   2975#   2988#   3005#   3024#   3041#   3054#   3075#   3101#
                                 3114#   3126#   3139#   3151#   3176#   3189#   3209#   3223#   3238#   3266#   3294#   3308#   3337#
                                 3364#   3391#   3426#   3438#   3476#   3524#   3543#   3563#   3583#   3611#   3630#   3671#   3698#
                                 3711#   3731#   3758#   3770#   3782#   3794#   3806#   3833#   3853#   3895#   3907#   3946#   3970#
                                 3982#   4722#   4846#   4863#   4905#   4915#   4926#   4947#   4967#   4995#   5051#   5059#   5077#
                                 5095#   5101#   5154#   5162#   5180#   5197#   5214#   5231#   5249#   5255#   5303#   5326#   5356#
                                 5405#   5414#   5442#   5468#   5497#   5518#   5537#   5557#   5560#   5587#   5607#   5629#   5645#
                                 5661#   5711#   5733#   5760#   5778#   5781#   5866#   6229#   6254#   6272#   6288#   6320#   6355#
                                 6358#   6406#   6451#   6468#   6494#   6512#   6520#   6555#   6586#   6597#   6641#   6681#   6701#
                                 6728#   6751#   6790#   6827#   6867#   6936#   6970#   7002#   7056#   7100#   7162#   7218#   7247#
                                 7282#   7329#   7383#   7428#   7476#   7562#   7593#   7646#   7715#   7752#   7800#   7874#   7905#
                                 7946#   8006#   8040#   8086#   8143#   8178#   8224#   8281#   8326#   8374#   8444#   8488#   8535#
```

```
                               8605#   8641#   8693#   8764#   8818#   8864#   8988#   9045#   9096#   9180#   9214#   9265#   9350#
                               9387#   9433#   9494#   9552#   9607#   9689#   9741#   9792#   9872#   9904#   9959#  10033#  10143#
                              10205#  10294#  10366#  10416#  10483#  10521#  10566#  10629#  10681#  10732#  10785#  10808#  10848#
                              10892#  10961#  11020#  11068#  11142#  11188#  11241#  11323#  11377#  11428#  11438#  11525#  11585#
                              11634#  11745#  11800#  11845#  11942#  12036#  12091#  12167#  12241#  12315#  12379#  12425#  12492#
                              12566#  12626#  12680#  12766#  13015#  13154#
TDRCNT  004001RG     002       2296#  12400   12410   12434   12448   12457   12471   12481   12501   12515   12524   12539   12552
TIMOFF  017302RG     002       4245#   4280    4283    4465    4468    4889    7187    7199
TIMON   017264RG     002       4232#   4273    4458    7179
TIMTST  002216RG     002       2132#   7303    7313    7335    7347    7368    7375
TLNKAD  003165RG     002       2223#   9925    9935    9965    9979    9988   10002   10016
TRAP4   022034RG     002       4943#   5017    5120    5272    5320
TRNDON  002445RG     002       2159#   8061    8071    8092    8106    8113    8122    8135
TXI    = 010000 G              1711#   4358    4361
TXIB   = 000020 G              1712#   4360
TSARGC= 000002                 1430#   1431#   1432#   1433#   1434#   1435#   2917#   2922    2930#   2937    2946#   2951    2953#
                               2959    2967#   2971    2980#   2984    2993#   3001    3010#   3014    3016#   3021    3029#   3037
                               3046#   3050    3059#   3063    3065#   3072    3080#   3084    3086#   3091    3093#   3098    3106#
                               3110    3119#   3123    3131#   3136    3144#   3148    3156#   3160    3162#   3166    3168#   3173
                               3181#   3186    3194#   3199    3201#   3206    3214#   3220    3228#   3235    3243#   3248    3250#
                               3255    3257#   3263    3271#   3276    3278#   3283    3285#   3291    3299#   3305    3313#   3318
                               3320#   3326    3328#   3334    3342#   3346    3348#   3353    3355#   3361    3369#   3373    3375#
                               3380    3382#   3388    3397#   3402    3405#   3412    3415#   3422    3431#   3435    3443#   3447
                               3449#   3453    3458#   3463    3465#   3470    3483#   3487    3490#   3494    3496#   3500    3502#
                               3507    3509#   3514    3516#   3521    3529#   3533    3535#   3540    3548#   3552    3554#   3560
                               3568#   3572    3574#   3580    3588#   3594    3596#   3601    3603#   3608    3616#   3620    3622#
                               3627    3635#   3639    3641#   3646    3648#   3653    3655#   3660    3662#   3668    3676#   3680
                               3682#   3688    3690#   3695    3703#   3708    3716#   3721    3723#   3728    3736#   3741    3743#
                               3747    3749#   3755    3763#   3767    3775#   3779    3787#   3791    3799#   3803    3811#   3815
                               3817#   3822    3824#   3830    3838#   3842    3844#   3850    3858#   3863    3865#   3870    3872#
                               3877    3879#   3884    3886#   3892    3900#   3904    3914#   3918    3922#   3926    3930#   3934
                               3938#   3942    3951#   3955    3957#   3961    3963#   3967    3975#   3979    4386#   4391    4422#
                               4428    4539#   4547    4554#   4558    4575#   4580    4641#   4646    4656#   4661    4669#   4674
                               4684#   4689    4697#   4702    4796#   4800    4882#   4886   12977#  12982   12984#  12989   12991#
                              12996   12998#  13003   13005#  13010
TSCODE= 001021                13142#  13147#
TSERRN= 000464                 1416#   5035#   5069#   5086#   5138#   5172#   5189#   5206#   5223#   5240#   5284#   5334#   5396#
                               5425#   5449#   5480#   5504#   5529#   5548#   5596#   5636#   5652#   5705#   5727#   5752#   5769#
                               5806#   5823#   5847#   5860#   6243#   6263#   6309#   6329#   6349#   6440#   6460#   6483#   6503#
                               6544#   6561#   6578#   6670#   6690#   6717#   6737#   6765#   6775#   6844#   6852#   6872#   6893#
                               6900#   6920#   6927#   6991#   7019#   7035#   7050#   7121#   7131#   7167#   7195#   7207#   7234#
                               7302#   7312#   7334#   7346#   7367#   7374#   7448#   7458#   7481#   7502#   7511#   7534#   7541#
                               7553#   7613#   7623#   7651#   7666#   7675#   7692#   7705#   7772#   7782#   7805#   7819#   7828#
                               7847#   7861#   7922#   7930#   7952#   7966#   7975#   7992#   8060#   8070#   8091#   8105#   8112#
                               8121#   8134#   8198#   8208#   8229#   8243#   8250#   8259#   8272#   8346#   8356#   8379#   8398#
                               8407#   8425#   8435#   8508#   8518#   8540#   8559#   8568#   8586#   8596#   8661#   8671#   8702#
                               8716#   8725#   8743#   8753#   8838#   8848#   8869#   8887#   8896#   8917#   8930#   8939#   8951#
                               8960#   8969#   8979#   9065#   9075#   9105#   9119#   9128#   9151#   9165#   9234#   9244#   9274#
                               9288#   9297#   9321#   9335#   9407#   9417#   9438#   9453#   9462#   9475#   9485#   9572#   9582#
                               9615#   9629#   9638#   9658#   9672#   9761#   9771#   9801#   9815#   9824#   9839#   9853#   9924#
                               9934#   9964#   9978#   9987#  10001#  10015#  10163#  10173#  10215#  10229#  10238#  10255#  10271#
                              10285#  10386#  10396#  10425#  10448#  10456#  10467#  10541#  10551#  10571#  10587#  10596#  10610#
                              10620#  10701#  10711#  10737#  10752#  10761#  10774#  10790#  10868#  10878#  10897#  10919#  10928#
                              10942#  10952#  11040#  11050#  11076#  11095#  11104#  11117#  11127#  11208#  11218#  11246#  11265#
                              11274#  11290#  11298#  11308#  11397#  11407#  11449#  11463#  11472#  11486#  11496#  11605#  11615#
                              11648#  11662#  11671#  11688#  11703#  11714#  11728#  11820#  11830#  11859#  11873#  11882#  11896#
```

```
                         11911#  11922#  11936#  12056#  12066#  12100#  12114#  12123#  12145#  12156#  12177#  12191#  12197#
                         12217#  12228#  12249#  12263#  12272#  12296#  12306#  12399#  12409#  12433#  12447#  12456#  12470#
                         12480#  12500#  12514#  12523#  12538#  12551#  12646#  12656#  12685#  12699#  12708#  12721#  12752#
                         12813#  12826#  12847#  12860#  12891#  12904#  12936#  12948#
TSEXCP= 000000           13142#  13146   13147#  13151
TSFLAG= 000040            4718#   4720    4891#   4911#   4913    4922#   4924    5400#   5756#   5810#   5851#   6247#   6313#
                          6444#   6487#   6548#   6674#   6694#   6721#   6741#   6876#   7023#   7109#   7125#   7135#   7171#
                          7291#   7306#   7316#   7338#   7350#   7437#   7452#   7462#   7485#   7506#   7515#   7602#   7617#
                          7627#   7655#   7670#   7679#   7709#   7761#   7776#   7786#   7809#   7823#   7832#   7865#   7956#
                          7970#   8049#   8064#   8074#   8095#   8116#   8125#   8187#   8202#   8212#   8233#   8254#   8263#
                          8335#   8350#   8360#   8383#   8402#   8411#   8497#   8512#   8522#   8544#   8563#   8572#   8650#
                          8665#   8675#   8706#   8720#   8729#   8827#   8842#   8852#   8873#   8891#   8900#   9054#   9069#
                          9079#   9109#   9123#   9132#   9169#   9223#   9238#   9248#   9278#   9292#   9301#   9339#   9396#
                          9411#   9421#   9442#   9457#   9466#   9561#   9576#   9586#   9619#   9633#   9642#   9676#   9750#
                          9765#   9775#   9805#   9819#   9828#   9857#   9913#   9928#   9938#   9968#   9982#   9991#  10019#
                         10152#  10167#  10177#  10219#  10233#  10242#  10375#  10390#  10400#  10429#  10452#  10530#  10545#
                         10555#  10575#  10591#  10600#  10690#  10705#  10715#  10741#  10756#  10765#  10794#  10857#  10872#
                         10882#  10901#  10923#  10932#  11029#  11044#  11054#  11080#  11099#  11108#  11197#  11212#  11222#
                         11250#  11269#  11278#  11386#  11401#  11411#  11453#  11467#  11476#  11594#  11609#  11619#  11652#
                         11666#  11675#  11732#  11809#  11824#  11834#  11863#  11877#  11886#  12045#  12060#  12070#  12104#
                         12118#  12127#  12181#  12201#  12253#  12267#  12276#  12388#  12403#  12413#  12437#  12451#  12460#
                         12504#  12518#  12527#  12542#  12635#  12650#  12660#  12689#  12703#  12712#  12725#  12756#  12790#
                         12796#  12817#  12830#  12851#  12864#  12895#  12908#  12940#  12952#
TSGMAN= 000000            1416#
TSHILI= 000776           13142#  13145   13147#  13150
TSLAST= 000000            1416#
TSLOLI= 000000           13142#  13144   13147#  13149
TSLSYM= 010000            1416#   1591    2925    2941    2962    2975    2988    3005    3024    3041    3054    3075    3101
                          3114    3126    3139    3151    3176    3189    3209    3223    3238    3266    3294    3308    3337
                          3364    3391    3426    3438    3476    3524    3543    3563    3583    3611    3630    3671    3698
                          3711    3731    3758    3770    3782    3794    3806    3833    3853    3895    3907    3946    3970
                          3982    4722    4846    4863    4905    4915    4926    4947    4967    4995    5051    5095    5101
                          5154    5249    5255    5303    5356    5405    5557    5560    5607    5661    5711    5733    5778
                          5781    5866    6272    6355    6358    6512    6586    6597    6790    6936    7056    7247    7383
                          7562    7715    7874    8006    8143    8281    8444    8605    8764    8988    9180    9350    9494
                          9689    9872   10033   10294   10483   10629   10808   10961   11142   11323   11525   11745   11942
                         12315   12566   12766   13015   13154
TSNEST= 177777            1416#   1420#   1582#   1590#   2915#   2924#   2928#   2940#   2944#   2961#   2965#   2974#   2978#
                          2987#   2991#   3004#   3008#   3023#   3027#   3040#   3044#   3053#   3057#   3074#   3078#   3100#
                          3104#   3113#   3117#   3125#   3129#   3138#   3142#   3150#   3154#   3175#   3179#   3188#   3192#
                          3208#   3212#   3222#   3226#   3237#   3241#   3265#   3269#   3293#   3297#   3307#   3311#   3336#
                          3340#   3363#   3367#   3390#   3394#   3425#   3429#   3437#   3441#   3475#   3479#   3523#   3527#
                          3542#   3546#   3562#   3566#   3582#   3586#   3610#   3614#   3629#   3633#   3670#   3674#   3697#
                          3701#   3710#   3714#   3730#   3734#   3757#   3761#   3769#   3773#   3781#   3785#   3793#   3797#
                          3805#   3809#   3832#   3836#   3852#   3856#   3894#   3898#   3906#   3910#   3945#   3949#   3969#
                          3973#   3981#   4716#   4721#   4733#   4740#   4749#   4845#   4860#   4862#   4872#   4904#   4909#
                          4914#   4920#   4925#   4943#   4946#   4962#   4966#   4988#   4994#   5015#   5024#   5050#   5054#
                          5059#   5074#   5077#   5091#   5094#   5100#   5118#   5127#   5153#   5157#   5162#   5177#   5180#
                          5194#   5197#   5211#   5214#   5228#   5231#   5245#   5248#   5254#   5270#   5302#   5318#   5326#
                          5349#   5355#   5377#   5382#   5404#   5409#   5414#   5430#   5442#   5454#   5468#   5485#   5497#
                          5509#   5518#   5534#   5537#   5553#   5556#   5559#   5582#   5587#   5601#   5606#   5626#   5629#
                          5641#   5645#   5657#   5660#   5689#   5691#   5710#   5714#   5732#   5736#   5760#   5774#   5777#
                          5780#   5802#   5865#   6222#   6224#   6229#   6251#   6254#   6268#   6271#   6275#   6288#   6317#
                          6320#   6334#   6354#   6357#   6398#   6401#   6406#   6448#   6451#   6465#   6468#   6491#   6494#
                          6508#   6511#   6515#   6520#   6552#   6555#   6566#   6585#   6596#   6637#   6641#   6678#   6681#
                          6698#   6701#   6725#   6728#   6745#   6751#   6780#   6789#   6821#   6827#   6858#   6867#   6932#
```

```
                         6935#   6965#   6970#   6996#   7002#   7040#   7055#   7094#   7100#   7145#   7162#   7212#   7218#
                         7239#   7246#   7276#   7282#   7320#   7329#   7379#   7382#   7422#   7428#   7466#   7476#   7558#
                         7561#   7587#   7593#   7631#   7646#   7696#   7714#   7746#   7752#   7790#   7800#   7852#   7873#
                         7899#   7905#   7936#   7946#   8002#   8005#   8034#   8040#   8078#   8086#   8139#   8142#   8172#
                         8178#   8216#   8224#   8277#   8280#   8320#   8326#   8364#   8374#   8440#   8443#   8482#   8488#
                         8526#   8535#   8601#   8604#   8635#   8641#   8679#   8693#   8758#   8763#   8812#   8818#   8856#
                         8864#   8984#   8987#   9039#   9045#   9083#   9096#   9156#   9179#   9208#   9214#   9252#   9265#
                         9326#   9349#   9381#   9387#   9425#   9433#   9490#   9493#   9546#   9552#   9590#   9607#   9663#
                         9688#   9735#   9741#   9779#   9792#   9844#   9871#   9898#   9904#   9942#   9959#  10006#  10032#
                        10137#  10143#  10181#  10205#  10290#  10293#  10360#  10366#  10404#  10416#  10472#  10482#  10515#
                        10521#  10559#  10566#  10625#  10628#  10675#  10681#  10719#  10732#  10779#  10785#  10798#  10807#
                        10842#  10848#  10886#  10892#  10957#  10960#  11014#  11020#  11058#  11068#  11133#  11141#  11182#
                        11188#  11226#  11241#  11313#  11322#  11371#  11377#  11415#  11428#  11438#  11501#  11514#  11524#
                        11579#  11585#  11623#  11634#  11719#  11744#  11794#  11800#  11836#  11845#  11927#  11941#  12030#
                        12036#  12074#  12091#  12161#  12167#  12235#  12241#  12311#  12314#  12373#  12379#  12417#  12425#
                        12485#  12492#  12556#  12565#  12620#  12626#  12664#  12680#  12762#  12765#  12787#  13014#  13139#
                        13152#  13179#
T$NSO = 000000           1420#  13179
T$NS1 = 000004           1582#   1590    2915#   2924    2928#   2940    2944#   2961    2965#   2974    2978#   2987    2991#
                         3004    3008#   3023    3027#   3040    3044#   3053    3057#   3074    3078#   3100    3104#   3113
                         3117#   3125    3129#   3138    3142#   3150    3154#   3175    3179#   3188    3192#   3208    3212#
                         3222    3226#   3237    3241#   3265    3269#   3293    3297#   3307    3311#   3336    3340#   3363
                         3367#   3390    3394#   3425    3429#   3437    3441#   3475    3479#   3523    3527#   3542    3546#
                         3562    3566#   3582    3586#   3610    3614#   3629    3633#   3670    3674#   3697    3701#   3710
                         3714#   3730    3734#   3757    3761#   3769    3773#   3781    3785#   3793    3797#   3805    3809#
                         3832    3836#   3852    3856#   3894    3898#   3906    3910#   3945    3949#   3969    3973#   3981
                         4716#   4721    4733#   4740    4749#   4845    4860#   4862    4872#   4904    4909#   4914    4920#
                         4925    4943#   4946    4962#   4966    4988#   4994    5015#   5100    5118#   5254    5270#   5302
                         5318#   5355    5377#   5559    5582#   5606    5626#   5660    5689#   5780    5802#   5865    6222#
                         6357    6398#   6596    6637#   6789    6821#   6935    6965#   7055    7094#   7246    7276#   7382
                         7422#   7561    7587#   7714    7746#   7873    7899#   8005    8034#   8142    8172#   8280    8320#
                         8443    8482#   8604    8635#   8763    8812#   8987    9039#   9179    9208#   9349    9381#   9493
                         9546#   9688    9735#   9871    9898#  10032   10137#  10293   10360#  10482   10515#  10628   10675#
                        10807   10842#  10960   11014#  11141   11182#  11322   11371#  11524   11579#  11744   11794#  11941
                        12030#  12314   12373#  12565   12620#  12765   12787#  13014   13139#  13152
T$NS2 = 000003           5024#   5050    5054#   5094    5127#   5153    5157#   5248    5326#   5349    5382#   5404    5409#
                         5556    5587#   5601    5629#   5641    5645#   5657    5691#   5710    5714#   5732    5736#   5777
                         6224#   6271    6275#   6354    6401#   6511    6515#   6585    6641#   6678    6681#   6698    6701#
                         6725    6728#   6745    6751#   6780    6827#   6858    6867#   6932    6970#   6996    7002#   7040
                         7100#   7145    7162#   7212    7218#   7239    7282#   7320    7329#   7379    7428#   7466    7476#
                         7558    7593#   7631    7646#   7696    7752#   7790    7800#   7852    7905#   7936    7946#   8002
                         8040#   8078    8086#   8139    8178#   8216    8224#   8277    8326#   8364    8374#   8440    8488#
                         8526    8535#   8601    8641#   8679    8693#   8758    8818#   8856    8864#   8984    9045#   9083
                         9096#   9156    9214#   9252    9265#   9326    9387#   9425    9433#   9490    9552#   9590    9607#
                         9663    9741#   9779    9792#   9844    9904#   9942    9959#  10006   10143#  10181   10205#  10290
                        10366#  10404   10416#  10472   10521#  10559   10566#  10625   10681#  10719   10732#  10779   10785#
                        10798   10848#  10886   10892#  10957   11020#  11058   11068#  11133   11188#  11226   11241#  11313
                        11377#  11415   11428#  11514   11585#  11623   11634#  11719   11800#  11836   11845#  11927   12036#
                        12074   12091#  12161   12167#  12235   12241#  12311   12379#  12417   12425#  12485   12492#  12556
                        12626#  12664   12680#  12762
T$NS3 = 000003           5059#   5074    5077#   5091    5162#   5177    5180#   5194    5197#   5211    5214#   5228    5231#
                         5245    5414#   5430    5442#   5454    5468#   5485    5497#   5509    5518#   5534    5537#   5553
                         5760#   5774    6229#   6251    6254#   6268    6288#   6317    6320#   6334    6406#   6448    6451#
                         6465    6468#   6491    6494#   6508    6520#   6552    6555#   6566   11438#  11501
T$PTNU= 000000           1416#
T$SAVL= 177777           1416#
```

```
T$SEGL= 177777      1416#   5059#   5074#   5076    5077#   5091#   5093    5162#   5177#   5179    5180#   5194#   5196
                    5197#   5211#   5213    5214#   5228#   5230    5231#   5245#   5247    5326#   5349#   5351    5414#
                    5430#   5432    5442#   5454#   5456    5468#   5485#   5487    5497#   5509#   5511    5518#   5534#
                    5536    5537#   5553#   5555    5587#   5601#   5603    5629#   5641#   5643    5645#   5657#   5659
                    5760#   5774#   5776    6229#   6251#   6253    6254#   6268#   6270    6288#   6317#   6319    6320#
                    6334#   6336    6406#   6448#   6450    6451#   6465#   6467    6468#   6491#   6493    6494#   6508#
                    6510    6520#   6552#   6554    6555#   6566#   6568    6641#   6678#   6680    6681#   6698#   6700
                    6701#   6725#   6727    6728#   6745#   6747    6751#   6780#   6782    6827#   6858#   6860    6867#
                    6932#   6934    6970#   6996#   6998    7002#   7040#   7042    7100#   7145#   7147    7162#   7212#
                    7214    7218#   7239#   7241    7282#   7320#   7322    7329#   7379#   7381    7428#   7466#   7468
                    7476#   7558#   7560    7593#   7631#   7633    7646#   7696#   7698    7752#   7790#   7792    7800#
                    7852#   7854    7905#   7936#   7938    7946#   8002#   8004    8040#   8078#   8080    8086#   8139#
                    8141    8178#   8216#   8218    8224#   8277#   8279    8326#   8364#   8366    8374#   8440#   8442
                    8488#   8526#   8528    8535#   8601#   8603    8641#   8679#   8681    8693#   8758#   8760    8818#
                    8856#   8858    8864#   8984#   8986    9045#   9083#   9085    9096#   9156#   9158    9214#   9252#
                    9254    9265#   9326#   9328    9387#   9425#   9427    9433#   9490#   9492    9552#   9590#   9592
                    9607#   9663#   9665    9741#   9779#   9781    9792#   9844#   9846    9904#   9942#   9944    9959#
                    10006#  10008   10143#  10181#  10183   10205#  10290#  10292   10366#  10404#  10406   10416#  10472#
                    10474   10521#  10559#  10561   10566#  10625#  10627   10681#  10719#  10721   10732#  10779#  10781
                    10785#  10798#  10800   10848#  10886#  10888   10892#  10957#  10959   11020#  11058#  11060   11068#
                    11133#  11135   11188#  11226#  11228   11241#  11313#  11315   11377#  11415#  11417   11428#  11438#
                    11501#  11503   11514#  11516   11585#  11623#  11625   11634#  11719#  11721   11800#  11838#  11840
                    11845#  11927#  11929   12036#  12074#  12076   12091#  12161#  12163   12167#  12235#  12237   12241#
                    12311#  12313   12379#  12417#  12419   12425#  12485#  12487   12492#  12556#  12558   12626#  12664#
                    12666   12680#  12762#  12764
T$SEKO= 010001      5059#   5074    5077#   5091    5162#   5177    5180#   5194    5197#   5211    5214#   5228    5231#
                    5245    5326#   5349    5414#   5430    5442#   5454    5468#   5485    5497#   5509    5518#   5534
                    5537#   5553    5587#   5601    5629#   5641    5645#   5657    5760#   5774    6229#   6251    6254#
                    6268    6288#   6317    6320#   6334    6406#   6448    6451#   6465    6468#   6491    6494#   6508
                    6520#   6552    6555#   6566    6641#   6678    6681#   6698    6701#   6725    6728#   6745    6751#
                    6780    6827#   6858    6867#   6932    6970#   6996    7002#   7040    7100#   7145    7162#   7212
                    7218#   7239    7282#   7320    7329#   7379    7428#   7466    7476#   7558    7593#   7631    7646#
                    7696    7752#   7790    7800#   7852    7905#   7936    7946#   8002    8040#   8078    8086#   8139
                    8178#   8216    8224#   8277    8326#   8364    8374#   8440    8488#   8526    8535#   8601    8641#
                    8679    8693#   8758    8818#   8856    8864#   8984    9045#   9083    9096#   9156    9214#   9252
                    9265#   9326    9387#   9425    9433#   9490    9552#   9590    9607#   9663    9741#   9779    9792#
                    9844    9904#   9942    9959#   10006   10143#  10181   10205#  10290   10366#  10404   10416#  10472
                    10521#  10559   10566#  10625   10681#  10719   10732#  10779   10785#  10798   10848#  10886   10892#
                    10957   11020#  11058   11068#  11133   11188#  11226   11241#  11313   11377#  11415   11428#  11514
                    11585#  11623   11634#  11719   11800#  11838   11845#  11927   12036#  12074   12091#  12161   12167#
                    12235   12241#  12311   12379#  12417   12425#  12485   12492#  12556   12626#  12664   12680#  12762
T$SEK1= 010002      11438#  11501
T$SUBN= 000000      1416#   5014#   5023#   5053#   5117#   5126#   5156#   5269#   5317#   5376#   5381#   5408#   5581#
                    5625#   5688#   5690#   5713#   5735#   5801#   6221#   6223#   6274#   6397#   6400#   6514#   6636#
                    6820#   6964#   7093#   7275#   7421#   7586#   7745#   7898#   8033#   8171#   8319#   8481#   8634#
                    8811#   9038#   9207#   9380#   9545#   9734#   9897#   10136#  10359#  10514#  10674#  10841#  11013#
                    11181#  11370#  11578#  11793#  12029#  12372#  12619#  12786#
T$TAGL= 177777      1416#
T$TAGN= 010172      1416#   1582#   2915#   2928#   2944#   2965#   2978#   2991#   3008#   3027#   3044#   3057#   3078#
                    3104#   3117#   3129#   3142#   3154#   3179#   3192#   3212#   3226#   3241#   3269#   3297#   3311#
                    3340#   3367#   3394#   3429#   3441#   3479#   3527#   3546#   3566#   3586#   3614#   3633#   3674#
                    3701#   3714#   3734#   3761#   3773#   3785#   3797#   3809#   3836#   3856#   3898#   3910#   3949#
                    3973#   4716#   4733#   4749#   4860#   4872#   4909#   4920#   4943#   4962#   4988#   5015#   5024#
                    5054#   5118#   5127#   5157#   5270#   5318#   5377#   5382#   5409#   5582#   5626#   5689#   5691#
                    5714#   5736#   5802#   6222#   6224#   6275#   6398#   6401#   6515#   6637#   6821#   6965#   7094#
                    7276#   7422#   7587#   7746#   7899#   8034#   8172#   8320#   8482#   8635#   8812#   9039#   9208#
```

```
                          9381#  9546#  9735#  9898#  10137# 10360# 10515# 10675# 10842# 11014# 11182# 11371# 11579#
                          11794# 12030# 12373# 12620# 12787# 13139#
TSTEMP= 000000            1525#  1526#  1527#  1528#  1529#  1530#  1531#  1532#  1533#  1534#  1535#  1536#  1537#
                          1538#  1539#  1540#  1541#  1542#  1543#  1544#  1545#  1546#  1547#  1548#  1549#  1550#
                          1551#  1552#  1553#  1554#  1555#  1556#  1557#  1558#  1559#  1560#  1561#  1562#  1563#
                          1564#  1565#  1566#  1567#  1568#  1569#  1570#  1571#  1590#  2924#  2940#  2961#  2974#
                          2987#  3004#  3023#  3040#  3053#  3074#  3100#  3113#  3125#  3138#  3150#  3175#  3188#
                          3208#  3222#  3237#  3265#  3293#  3307#  3336#  3363#  3390#  3425#  3437#  3475#  3523#
                          3542#  3562#  3582#  3610#  3629#  3670#  3697#  3710#  3730#  3757#  3769#  3781#  3793#
                          3805#  3832#  3852#  3894#  3906#  3945#  3969#  3981#  4718#  4719   4721#  4740#  4845#
                          4862#  4891#  4892   4904#  4911#  4912   4914#  4922#  4923   4925#  4946#  4966#  4994#
                          5050#  5074#  5091#  5094#  5100#  5153#  5177#  5194#  5211#  5228#  5245#  5248#  5254#
                          5302#  5349#  5355#  5400#  5401   5404#  5430#  5454#  5485#  5509#  5534#  5553#  5556#
                          5559#  5601#  5606#  5641#  5657#  5660#  5710#  5732#  5756#  5757   5774#  5777#  5780#
                          5810#  5811   5851#  5852   5865#  6247#  6248   6251#  6268#  6271#  6313#  6314   6317#
                          6334#  6354#  6357#  6444#  6445   6448#  6465#  6487#  6488   6491#  6508#  6511#  6548#
                          6549   6552#  6566#  6585#  6596#  6674#  6675   6678#  6694#  6695   6698#  6721#  6722
                          6725#  6741#  6742   6745#  6780#  6789#  6858#  6876#  6877   6932#  6935#  6996#  7023#
                          7024   7040#  7055#  7109#  7110   7125#  7126   7135#  7136   7145#  7171#  7172   7212#
                          7239#  7246#  7291#  7292   7306#  7307   7316#  7317   7320#  7338#  7339   7350#  7351
                          7379#  7382#  7437#  7438   7452#  7453   7462#  7463   7466#  7485#  7486   7506#  7507
                          7515#  7516   7558#  7561#  7602#  7603   7617#  7618   7627#  7628   7631#  7655#  7656
                          7670#  7671   7679#  7680   7696#  7709#  7710   7714#  7761#  7762   7776#  7777   7786#
                          7787   7790#  7809#  7810   7823#  7824   7832#  7833   7852#  7865#  7866   7873#  7936#
                          7956#  7957   7970#  7971   8002#  8005#  8049#  8050   8064#  8065   8074#  8075   8078#
                          8095#  8096   8116#  8117   8125#  8126   8139#  8142#  8187#  8188   8202#  8203   8212#
                          8213   8216#  8233#  8234   8254#  8255   8263#  8264   8277#  8280#  8335#  8336   8350#
                          8351   8360#  8361   8364#  8383#  8384   8402#  8403   8411#  8412   8440#  8443#  8497#
                          8498   8512#  8513   8522#  8523   8526#  8544#  8545   8563#  8564   8572#  8573   8601#
                          8604#  8650#  8651   8665#  8666   8675#  8676   8679#  8706#  8707   8720#  8721   8729#
                          8730   8758#  8763#  8827#  8828   8842#  8843   8852#  8853   8856#  8873#  8874   8891#
                          8892   8900#  8901   8984#  8987#  9054#  9055   9069#  9070   9079#  9080   9083#  9109#
                          9110   9123#  9124   9132#  9133   9156#  9169#  9170   9179#  9223#  9224   9238#  9239
                          9248#  9249   9252#  9278#  9279   9292#  9293   9301#  9302   9326#  9339#  9340   9349#
                          9396#  9397   9411#  9412   9421#  9422   9425#  9442#  9443   9457#  9458   9466#  9467
                          9490#  9493#  9561#  9562   9576#  9577   9586#  9587   9590#  9619#  9620   9633#  9634
                          9642#  9643   9663#  9676#  9677   9688#  9750#  9751   9765#  9766   9775#  9776   9779#
                          9805#  9806   9819#  9820   9828#  9829   9844#  9857#  9858   9871#  9913#  9914   9928#
                          9929   9938#  9939   9942#  9968#  9969   9982#  9983   9991#  9992   10006# 10019# 10020
                          10032# 10152# 10153  10167# 10168  10177# 10178  10181# 10219# 10220  10233# 10234  10242#
                          10243  10290# 10293# 10375# 10376  10390# 10391  10400# 10401  10404# 10429# 10430  10452#
                          10453  10472# 10482# 10530# 10531  10545# 10546  10555# 10556  10559# 10575# 10576  10591#
                          10592  10600# 10601  10625# 10628# 10690# 10691  10705# 10706  10715# 10716  10719# 10741#
                          10742  10756# 10757  10765# 10766  10779# 10794# 10795  10798# 10807# 10857# 10858  10872#
                          10873  10882# 10883  10886# 10901# 10902  10923# 10924  10932# 10933  10957# 10960# 11029#
                          11030  11044# 11045  11054# 11055  11058# 11080# 11081  11099# 11100  11108# 11109  11133#
                          11141# 11197# 11198  11212# 11213  11222# 11223  11226# 11250# 11251  11269# 11270  11278#
                          11279  11313# 11322# 11386# 11387  11401# 11402  11411# 11412  11415# 11453# 11454  11467#
                          11468  11476# 11477  11501# 11514# 11524# 11594# 11595  11609# 11610  11619# 11620  11623#
                          11652# 11653  11666# 11667  11675# 11676  11719# 11732# 11733  11744# 11809# 11810  11824#
                          11825  11834# 11835  11838# 11863# 11864  11877# 11878  11886# 11887  11927# 11941# 12045#
                          12046  12060# 12061  12070# 12071  12074# 12104# 12105  12118# 12119  12127# 12128  12161#
                          12181# 12182  12201# 12202  12235# 12253# 12254  12267# 12268  12276# 12277  12311# 12314#
                          12388# 12389  12403# 12404  12413# 12414  12417# 12437# 12438  12451# 12452  12460# 12461
                          12485# 12504# 12505  12518# 12519  12527# 12528  12542# 12543  12556# 12565# 12635# 12636
                          12650# 12651  12660# 12661  12664# 12689# 12690  12703# 12704  12712# 12713  12725# 12726
```

```
                        12756#  12757   12762#  12765#  12790#  12791   12796#  12797   12817#  12818   12830#  12831   12851#
                        12852   12864#  12865   12895#  12896   12908#  12909   12940#  12941   12952#  12953   13014#  13142#
                        13147#  13152#  13179#
T$TEST= 000056           1416#   5014#   5023    5053    5117#   5126    5156    5269#   5317#   5376#   5381    5408    5581#
                         5625#   5688#   5690    5713    5735    5801#   6221#   6223    6274    6397#   6400    6514    6636#
                         6820#   6964#   7093#   7275#   7421#   7586#   7745#   7898#   8033#   8171#   8319#   8481#   8634#
                         8811#   9038#   9207#   9380#   9545#   9734#   9897#  10136#  10359#  10514#  10674#  10841#  11013#
                        11181#  11370#  11578#  11793#  12029#  12372#  12619#  12786#
T$TSTM= 177777           1416#   2921    2925    2936    2941    2950    2958    2962    2970    2975    2983    2988    3000
                         3005    3013    3020    3024    3036    3041    3049    3054    3062    3071    3075    3083    3090
                         3097    3101    3109    3114    3122    3126    3135    3139    3147    3151    3159    3165    3172
                         3176    3185    3189    3198    3205    3209    3219    3223    3234    3238    3247    3254    3262
                         3266    3275    3282    3290    3294    3304    3308    3317    3325    3333    3337    3345    3352
                         3360    3364    3372    3379    3387    3391    3401    3411    3421    3426    3434    3438    3446
                         3452    3462    3469    3476    3486    3493    3499    3506    3513    3520    3524    3532    3539
                         3543    3551    3559    3563    3571    3579    3583    3593    3600    3607    3611    3619    3626
                         3630    3638    3645    3652    3659    3667    3671    3679    3687    3694    3698    3707    3711
                         3720    3727    3731    3740    3746    3754    3758    3766    3770    3778    3782    3790    3794
                         3802    3806    3814    3821    3829    3833    3841    3849    3853    3862    3869    3876    3883
                         3891    3895    3903    3907    3917    3925    3933    3941    3946    3954    3960    3966    3970
                         3978    3982    4234    4248    4277    4390    4427    4462    4546    4557    4579    4645    4660
                         4673    4688    4701    4722    4755    4760    4775    4780    4785    4791    4799    4813    4823
                         4842    4846    4863    4885    4891    4905    4915    4926    5020    5024    5034    5039    5042
                         5045    5047    5051    5054    5059    5068    5075    5077    5085    5092    5095    5098    5101
                         5123    5127    5137    5142    5145    5148    5150    5154    5157    5162    5171    5178    5180
                         5188    5195    5197    5205    5212    5214    5222    5229    5231    5239    5246    5249    5252
                         5255    5275    5283    5288    5291    5294    5296    5299    5303    5323    5326    5333    5338
                         5341    5344    5346    5350    5353    5356    5382    5395    5400    5405    5409    5414    5424
                         5431    5442    5448    5455    5468    5479    5486    5497    5503    5510    5518    5528    5535
                         5537    5547    5554    5557    5560    5587    5595    5602    5607    5629    5635    5642    5645
                         5651    5658    5661    5691    5704    5711    5714    5726    5733    5736    5751    5756    5760
                         5768    5775    5778    5781    5805    5810    5822    5846    5851    5859    5866    6224    6229
                         6242    6247    6252    6254    6262    6269    6272    6275    6288    6308    6313    6318    6320
                         6328    6335    6348    6355    6358    6401    6406    6439    6444    6449    6451    6459    6466
                         6468    6482    6487    6492    6494    6502    6509    6512    6515    6520    6543    6548    6553
                         6555    6560    6567    6577    6586    6597    6641    6669    6674    6679    6681    6689    6694
                         6699    6701    6716    6721    6726    6728    6736    6741    6746    6751    6764    6774    6781
                         6790    6827    6843    6851    6859    6867    6871    6876    6892    6899    6919    6926    6933
                         6936    6970    6976    6990    6997    7002    7009    7018    7023    7034    7041    7044    7049
                         7056    7100    7109    7120    7125    7130    7135    7142    7146    7162    7166    7171    7182
                         7194    7206    7213    7218    7233    7240    7247    7282    7291    7301    7306    7311    7316
                         7321    7329    7333    7338    7345    7350    7366    7373    7380    7383    7428    7437    7447
                         7452    7457    7462    7467    7476    7480    7485    7501    7506    7510    7515    7533    7540
                         7552    7559    7562    7593    7602    7612    7617    7622    7627    7632    7646    7650    7655
                         7665    7670    7674    7679    7691    7697    7704    7709    7715    7752    7761    7771    7776
                         7781    7786    7791    7800    7804    7809    7818    7823    7827    7832    7846    7853    7860
                         7865    7874    7905    7921    7929    7937    7946    7951    7956    7965    7970    7974    7991
                         8003    8006    8040    8049    8059    8064    8069    8074    8079    8086    8090    8095    8104
                         8111    8116    8120    8125    8133    8140    8143    8178    8187    8197    8202    8207    8212
                         8217    8224    8228    8233    8242    8249    8254    8258    8263    8271    8278    8281    8326
                         8335    8345    8350    8355    8360    8365    8374    8378    8383    8397    8402    8406    8411
                         8424    8434    8441    8444    8488    8497    8507    8512    8517    8522    8527    8535    8539
                         8544    8558    8563    8567    8572    8585    8595    8602    8605    8641    8650    8660    8665
                         8670    8675    8680    8693    8701    8706    8715    8720    8724    8729    8742    8752    8759
                         8764    8818    8827    8837    8842    8847    8852    8857    8864    8868    8873    8886    8891
                         8895    8900    8916    8929    8938    8950    8959    8968    8978    8985    8988    9045    9054
```

```
                      9064    9069    9074    9079    9084    9096    9104    9109    9118    9123    9127    9132    9150
                      9157    9164    9169    9180    9214    9223    9233    9238    9243    9248    9253    9265    9273
                      9278    9287    9292    9296    9301    9320    9327    9334    9339    9350    9387    9396    9406
                      9411    9416    9421    9426    9433    9437    9442    9452    9457    9461    9466    9474    9484
                      9491    9494    9552    9561    9571    9576    9581    9586    9591    9607    9614    9619    9628
                      9633    9637    9642    9657    9664    9671    9676    9689    9741    9750    9760    9765    9770
                      9775    9780    9792    9800    9805    9814    9819    9823    9828    9838    9845    9852    9857
                      9872    9904    9913    9923    9928    9933    9938    9943    9959    9963    9968    9977    9982
                      9986    9991   10000   10007   10014   10019   10033   10143   10152   10162   10167   10172   10177
                     10182   10205   10214   10219   10228   10233   10237   10242   10254   10270   10284   10291   10294
                     10366   10375   10385   10390   10395   10400   10405   10416   10424   10429   10447   10452   10455
                     10466   10473   10483   10521   10530   10540   10545   10550   10555   10560   10566   10570   10575
                     10586   10591   10595   10600   10609   10619   10626   10629   10681   10690   10700   10705   10710
                     10715   10720   10732   10736   10741   10751   10756   10760   10765   10773   10780   10785   10789
                     10794   10799   10808   10848   10857   10867   10872   10877   10882   10887   10892   10896   10901
                     10918   10923   10927   10932   10941   10951   10958   10961   11020   11029   11039   11044   11049
                     11054   11059   11068   11075   11080   11094   11099   11103   11108   11116   11126   11134   11142
                     11188   11197   11207   11212   11217   11222   11227   11241   11245   11250   11264   11269   11273
                     11278   11289   11297   11307   11314   11323   11377   11386   11396   11401   11406   11411   11416
                     11428   11438   11448   11453   11462   11467   11471   11476   11485   11495   11502   11515   11525
                     11585   11594   11604   11609   11614   11619   11624   11634   11647   11652   11661   11666   11670
                     11675   11687   11702   11713   11720   11727   11732   11745   11800   11809   11819   11824   11829
                     11834   11839   11845   11858   11863   11872   11877   11881   11886   11895   11910   11921   11928
                     11935   11942   12036   12045   12055   12060   12065   12070   12075   12091   12099   12104   12113
                     12118   12122   12127   12144   12155   12162   12167   12176   12181   12190   12196   12201   12216
                     12227   12236   12241   12248   12253   12262   12267   12271   12276   12295   12305   12312   12315
                     12379   12388   12398   12403   12408   12413   12418   12425   12432   12437   12446   12451   12455
                     12460   12469   12479   12486   12492   12499   12504   12513   12518   12522   12527   12537   12542
                     12550   12557   12566   12626   12635   12645   12650   12655   12660   12665   12680   12684   12689
                     12698   12703   12707   12712   12720   12725   12751   12756   12763   12766   12790   12796   12812
                     12817   12825   12830   12846   12851   12859   12864   12890   12895   12903   12908   12935   12940
                     12947   12952   12981   12988   12995   13002   13009   13015
TSTSTS= 000001        1416#   5015#   5118#   5270#   5318#   5377#   5582#   5626#   5689#   5802#   6222#   6398#   6637#
                      6821#   6965#   7094#   7276#   7422#   7587#   7746#   7899#   8034#   8172#   8320#   8482#   8635#
                      8812#   9039#   9208#   9381#   9546#   9735#   9898#  10137#  10360#  10515#  10675#  10842#  11014#
                     11182#  11371#  11579#  11794#  12030#  12373#  12620#  12787#
TS$AU = 010072        4920#   4922    4925
TS$AUT= 010067        4860#   4862
TS$CLE= 010070        4872#   4891    4904
TS$DU = 010071        4909#   4911    4914
TS$HAR= 010171       13139#  13153
TS$HW = 010000        1582#   1590
TS$INI= 010066        4749#   4845
TS$MSG= 010063        2915#   2924    2928#   2940    2944#   2961    2965#   2974    2978#   2987    2991#   3004    3008#
                      3023    3027#   3040    3044#   3053    3057#   3074    3078#   3100    3104#   3113    3117#   3125
                      3129#   3138    3142#   3150    3154#   3175    3179#   3188    3192#   3208    3212#   3222    3226#
                      3237    3241#   3265    3269#   3293    3297#   3307    3311#   3336    3340#   3363    3367#   3390
                      3394#   3425    3429#   3437    3441#   3475    3479#   3523    3527#   3542    3546#   3562    3566#
                      3582    3586#   3610    3614#   3629    3633#   3670    3674#   3697    3701#   3710    3714#   3730
                      3734#   3757    3761#   3769    3773#   3781    3785#   3793    3797#   3805    3809#   3832    3836#
                      3852    3856#   3894    3898#   3906    3910#   3945    3949#   3969    3973#   3981
TS$PRO= 010065        4733#
TS$RPT= 010064        4716#   4718    4721
TS$SEG= 010001        5059#   5074#   5077#   5091#   5162#   5177#   5180#   5194#   5197#   5211#   5214#   5228#   5231#
                      5245#   5326#   5349#   5414#   5430#   5442#   5454#   5468#   5485#   5497#   5509#   5518#   5534#
                      5537#   5553#   5587#   5601#   5629#   5641#   5645#   5657#   5760#   5774#   6229#   6251#   6254#
```

```
                                6268#   6288#   6317#   6320#   6334#   6406#   6448#   6451#   6465#   6468#   6491#   6494#   6508#
                                6520#   6552#   6555#   6566#   6641#   6678#   6681#   6698#   6701#   6725#   6728#   6745#   6751#
                                6780#   6827#   6858#   6867#   6932#   6970#   6996#   7002#   7040#   7100#   7145#   7162#   7212#
                                7218#   7239#   7282#   7320#   7329#   7379#   7428#   7466#   7476#   7558#   7593#   7631#   7646#
                                7696#   7752#   7790#   7800#   7852#   7905#   7936#   7946#   8002#   8040#   8078#   8086#   8139#
                                8178#   8216#   8224#   8277#   8326#   8364#   8374#   8440#   8488#   8526#   8535#   8601#   8641#
                                8679#   8693#   8758#   8818#   8856#   8864#   8984#   9045#   9083#   9096#   9156#   9214#   9252#
                                9265#   9326#   9387#   9425#   9433#   9490#   9552#   9590#   9607#   9663#   9761#   9779#   9792#
                                9844#   9904#   9942#   9959#   10006#  10143#  10181#  10205#  10290#  10366#  10404#  10416#  10472#
                                10521#  10559#  10566#  10625#  10681#  10719#  10732#  10779#  10785#  10798#  10848#  10886#  10892#
                                10957#  11020#  11058#  11068#  11133#  11188#  11226#  11241#  11313#  11377#  11415#  11428#  11438#
                                11501#  11514#  11585#  11623#  11634#  11719#  11800#  11838#  11845#  11927#  12036#  12074#  12091#
                                12161#  12167#  12235#  12241#  12311#  12379#  12417#  12425#  12485#  12492#  12556#  12626#  12664#
                                12680#  12762#
TSSSRV= 010075                  4943#   4946    4962#   4966    4988#   4994
TSSSUB= 010125                  5024#   5050    5054#   5094    5127#   5153    5157#   5248    5382#   5404    5409#   5556    5691#
                                5710    5714#   5732    5736#   5777    6224#   6271    6275#   6354    6401#   6511    6515#   6585
TSSTES= 010170                  5015#   5100    5118#   5254    5270#   5302    5318#   5355    5377#   5400    5559    5582#   5606
                                5626#   5660    5689#   5756    5780    5802#   5810    5851    5865    6222#   6247    6313    6357
                                6398#   6444    6487    6548    6596    6637#   6674    6694    6721    6741    6789    6821#   6876
                                6935    6965#   7023    7055    7094#   7109    7125    7135    7171    7246    7276#   7291    7306
                                7316    7338    7350    7382    7422#   7437    7452    7462    7485    7506    7515    7561    7587#
                                7602    7617    7627    7655    7670    7679    7709    7714    7746#   7761    7776    7786    7809
                                7823    7832    7865    7873    7899#   7956    7970    8005    8034#   8049    8064    8074    8095
                                8116    8125    8142    8172#   8187    8202    8212    8233    8254    8263    8280    8320#   8335
                                8350    8360    8383    8402    8411    8443    8482#   8497    8512    8522    8544    8563    8572
                                8604    8635#   8650    8665    8675    8706    8720    8729    8763    8812#   8827    8842    8852
                                8873    8891    8900    8987    9039#   9054    9069    9079    9109    9123    9132    9169    9179
                                9208#   9223    9238    9248    9278    9292    9301    9339    9349    9381#   9396    9411    9421
                                9442    9457    9466    9493    9546#   9561    9576    9586    9619    9633    9642    9676    9688
                                9735#   9750    9765    9775    9805    9819    9828    9857    9871    9898#   9913    9928    9938
                                9968    9982    9991    10019   10032   10137#  10152   10167   10177   10219   10233   10242   10293
                                10360#  10375   10390   10400   10429   10452   10482   10515#  10530   10545   10555   10575   10591
                                10600   10628   10675#  10690   10705   10715   10741   10756   10765   10794   10807   10842#  10857
                                10872   10882   10901   10923   10932   10960   11014#  11029   11044   11054   11080   11099   11108
                                11141   11182#  11197   11212   11222   11250   11269   11278   11322   11371#  11386   11401   11411
                                11453   11467   11476   11524   11579#  11594   11609   11619   11652   11666   11675   11732   11744
                                11794#  11809   11824   11834   11863   11877   11886   11941   12030#  12045   12060   12070   12104
                                12118   12127   12181   12201   12253   12267   12276   12314   12373#  12388   12403   12413   12437
                                12451   12460   12504   12518   12527   12542   12565   12620#  12635   12650   12660   12689   12703
                                12712   12725   12756   12765   12787#  12790   12796   12817   12830   12851   12864   12895   12908
                                12940   12952   13014
T1          022076RG    002     1525    5014#
T1.1        022124R     002     5023#
T1.2        022202R     002     5053#
T10         027124RG    002     1534    6221#
T10.1       027124R     002     6223#
T10.2       027256R     002     6274#
T11         027540RG    002     1535    6397#
T11.1       027544R     002     6400#   6593
T11.2       030076R     002     6514#
T12         030326RG    002     1536    6636#
T13         031006RG    002     1537    6820#
T14         031424RG    002     1538    6964#
T15         031714RG    002     1539    7093#
T16         032422RG    002     1540    7275#
```

```
T17       032740RG   002   1541     7421#
T18       033406RG   002   1542     7586#
T19       033764RG   002   1543     7745#
T2        022310RG   002   1526     5117#
T2.1      022336R    002   5126#
T2.2      022416R    002   5156#
T20       034344RG   002   1544     7898#
T21       034672RG   002   1545     8033#
T22       035202RG   002   1546     8171#
T23       035512RG   002   1547     8319#
T24       036072RG   002   1548     8481#
T25       036452RG   002   1549     8634#
T26       037046RG   002   1550     8811#
T27       037610RG   002   1551     9038#
T28       040206RG   002   1552     9207#
T29       040610RG   002   1553     9380#
T3        022642RG   002   1527     5269#
T30       041130RG   002   1554     9545#
T31       041520RG   002   1555     9734#
T32       042104RG   002   1556     9897#
T33       042470RG   002   1557    10136#
T34       043130RG   002   1558    10359#
T35       043460RG   002   1559    10514#
T36       044000RG   002   1560    10674#
T37       044350RG   002   1561    10841#
T38       044670RG   002   1562    11013#
T39       045230RG   002   1563    11181#
T4        022754RG   002   1528     5317#
T40       045620RG   002   1564    11370#
T41       046202RG   002   1565    11578#
T42       046634RG   002   1566    11793#
T43       047240RG   002   1567    12029#
T44       050330RG   002   1568    12372#
T45       051052RG   002   1569    12619#
T46       051466RG   002   1570    12786#
T5        023072RG   002   1529     5376#
T5.1      023104R    002   5381#
T5.2      023162R    002   5408#
T6        023560RG   002   1530     5581#
T7        023642RG   002   1531     5625#
T8        023740RG   002   1532     5688#
T8.1      023740R    002   5690#
T8.2      024004R    002   5713#
T8.3      024044R    002   5735#
T9        024162RG   002   1533     5801#
UAM     = 000200 G         1662#
UDBB      000616RG   002   1932#    4586*   4587*   4593*   4594*   4600*   4601*   4607*   4608*   4614*   4615*   4621*   4622*
                           4625*    4626*   4628*   4629*   4633    6281    6283*   6295*   6296*   6297*   6342*   6418*   6419*
                           6420*    6421*   6424    6531*   6532*   6647*   6648*   6649*   6650*   6654    12919   12921*  12922*
                           12923*   12924*
UNACSR    000266RG   002   1807#    4827*   4830
UNAINT    000672RG   002   1941#    4963*   6978*   7014    7173*   7183    7200
UNAPRI    000272RG   002   1809#    4829*   6972    7026    7138
UNASRV    022044RG   002   4962#    6973    7139
UNAVEC    000270RG   002   1808#    4828*   6974    7043    7140
UNIERR= 020000 G           1729#    3927    4496    4498
```

```
UNIHLT= 000005 G          1762#
UNIT    000330RG    002   1829#   4817*   4818*   4819    4822    5044    5147    5293    5343
UNLOD   001517RG    002   2075#   4576    4642    4657    4670    4685    4698
UNUSED  027112RG    002   5842    6177#
WCSADR= 000000 G          1683#   1684    4629    4707    6421    6650    6753
WCSMEM  002072RG    002   2117#   6845    6853    6873    6894    6901    6921    6928
WCSSIZ= 020000 G          1679#   1684    4629    4707    6412    6418    6421    6526    6572    6650    6753
WLSA  = 000025 G          1796#
WMAL  = 000007 G          1782#
WPA   = 000005 G          1780#
WRF   = 000011 G          1784#
WSIDP = 000023 G          1794#
WTM   = 000015 G          1788#
XPWR  = 100000 G          1739#   5541
X$ALWA= 000000            1416#
X$FALS= 000040            1416#
X$OFFS= 000400            1416#
X$TRUE= 000020            1416#
$ADRER  001461RG    002   2064#   6912    7528
$AFTER  001342RG    002   2046#   4536    5378    5493    5748    5819    6239    6305    6436    6479    6540    6666    6713
                          6761    6840    6987    7117    7298    7444    7609    7768    7918    8056    8194    8342    8504
                          8657    8834    9061    9230    9403    9568    9757    9920    10159   10382   10537   10697   10864
                          11036   11204   11393   11601   11816   12052   12395   12642   12809   12843   12887   12932
$BEFOR  001333RG    002   2044#
$BIT0   001271RG    002   1845    1880    2035#
$BIT1   001263RG    002   1846    1881    2033#
$BIT10  001174RG    002   1890    2015#
$BIT11  001165RG    002   1891    2013#
$BIT12  001156RG    002   1892    2011#
$BIT13  001147RG    002   1893    2009#   8927    8948
$BIT14  001140RG    002   1894    2007#   8957
$BIT15  001131RG    002   1895    2005#   8936    8966
$BIT2   001255RG    002   1847    1882    2031#
$BIT3   001247RG    002   1848    1883    2029#
$BIT4   001241RG    002   1849    1866    1884    2027#
$BIT5   001233RG    002   1867    1885    2025#
$BIT6   001225RG    002   1868    1886    2023#
$BIT7   001217RG    002   1887    2021#
$BIT8   001211RG    002   1888    2019#
$BIT9   001203RG    002   1854    1889    2017#
$CLR    001325RG    002   2042#
$DATER  001446RG    002   2062#   6908    7524
$DNI    001000RG    002   1856    1985#   4370    4534    5392    5746    5817    6237    6303    6434    6477    6538    6664
                          6711    6759    6838    6985    7115    7296    7442    7607    7766    7916    8054    8192    8340
                          8502    8655    8832    9059    9228    9401    9566    9755    9918    10157   10380   10535   10695
                          10862   11034   11202   11391   11599   11814   12050   12393   12640   12807   12841   12885   12930
$FATI   001021RG    002   1853    1989#   4384
$GTCMD  001357RG    002   2049#   6306    6480    6541    6714    6762    6841    7118    7299    7445    7610    7769    7919
                          8057    8195    8343    8505    8658    8835    9062    9231    9404    9569    9758    9921    10160
                          10383   10538   10698   10865   11037   11205   11394   11602   11817   12053   12396   12643   12844
                          12888   12933
$GTPCB  001350RG    002   2047#   6240    6437    6667    12810
$ICAB   001076RG    002   1876    1999#   5526
$INTE   001043RG    002   1851    1993#
$INTR   001032RG    002   1852    1991#   5421
$NCLR   001313RG    002   2040#   5063    5166    5439    5492    5525    5544
```

```
$NOP     001405RG   002   2054#   5749    6988
$NSET    001277RG   002   2037#   4535    5393    5422    5747    5818    6238    6304    6435    6478    6539    6665    6712
                          6760    6839    6986    7116    7157    7297    7443    7608    7767    7917    8055    8193    8341
                          8503    8656    8833    9060    9229    9402    9567    9756    9919    10158   10381   10536   10696
                          10863   11035   11203   11392   11600   11815   12051   12394   12641   12808   12842   12886   12931
$PARER   001502RG   002   2067#   6914
$PATCH   053366RG   002   13167#
$PCEI    000747RG   002   1859    1979#   4349
$PCTO    001107RG   002   1869    2001#
$PDNDM   001417RG   002   2057#
$RCEI    001010RG   002   1855    1987#   4377
$RESET   001440RG   002   2061#
$RMTC    001120RG   002   1865    2003#
$RSET    001054RG   002   1850    1995#
$RXI     000760RG   002   1858    1981#   4356
$SERI    000736RG   002   1860    1977#   4342
$SET     001307RG   002   2039#
$SLFT    001373RG   002   2052#   5820
$STOP    001433RG   002   2060#
$STRT    001411RG   002   2055#
$TXI     000770RG   002   1857    1983#   4363
$XPWR    001065RG   002   1877    1997#   5545
.      = 000000R    012   1937#   2900#   4719    4892    4901#   4912    4923    5401    5757    5811    5852    6248    6314
                          6445    6488    6549    6675    6695    6722    6742    6877    7024    7110    7126    7136    7172
                          7292    7307    7317    7339    7351    7438    7453    7463    7486    7507    7516    7603    7618
                          7628    7656    7671    7680    7710    7762    7777    7787    7810    7824    7833    7866    7957
                          7971    8050    8065    8075    8096    8117    8126    8188    8203    8213    8234    8255    8264
                          8336    8351    8361    8384    8403    8412    8498    8513    8523    8545    8564    8573    8651
                          8666    8676    8707    8721    8730    8828    8843    8853    8874    8892    8901    9055    9070
                          9080    9110    9124    9133    9170    9224    9239    9249    9279    9293    9302    9340    9397
                          9412    9422    9443    9458    9467    9562    9577    9587    9620    9634    9643    9677    9751
                          9766    9776    9806    9820    9829    9858    9914    9929    9939    9969    9983    9992    10020
                          10153   10168   10178   10220   10234   10243   10376   10391   10401   10430   10453   10531   10546
                          10556   10576   10592   10601   10691   10706   10716   10742   10757   10766   10795   10858   10873
                          10883   10902   10924   10933   11030   11045   11055   11081   11100   11109   11198   11213   11223
                          11251   11270   11279   11387   11402   11412   11454   11468   11477   11595   11610   11620   11653
                          11667   11676   11733   11810   11825   11835   11864   11878   11887   12046   12061   12071   12105
                          12119   12128   12182   12202   12254   12268   12277   12389   12404   12414   12438   12452   12461
                          12505   12519   12528   12543   12636   12651   12661   12690   12704   12713   12726   12757   12791
                          12797   12818   12831   12852   12865   12896   12909   12941   12953   13065#  13166#  13167#
```

```
BCOMPL       1#    1416#   4756    4793
BERROR       1#    1416#
BGNAU        1#    1416#   4919
BGNAUT       1#    1416#   4859
BGNCLN       1#    1416#   4871
BGNDU        1#    1416#   4908
BGNHRD       1#    1416#  13138
BGNHW        1#    1416#   1581
BGNINI       1#    1416#   4748
BGNMOD       1#    1416#   1419
BGNMSG       1#    1416#   2914    2927    2943    2964    2977    2990    3007    3026    3043    3056    3077    3103    3116
          3128     3141    3153    3178    3191    3211    3225    3240    3268    3296    3310    3339    3366    3393    3428
          3440     3478    3526    3545    3565    3585    3613    3632    3673    3700    3713    3733    3760    3772    3784
          3796     3808    3835    3855    3897    3909    3948    3972
BGNPRO       1#    1416#   4732
BGNPTA       1#    1416#
BGNRPT       1#    1416#   4715
BGNSEG       1#    1416#   5058    5076    5161    5179    5196    5213    5230    5325    5413    5441    5467    5496    5517
          5536     5586    5628    5644    5759    6228    6253    6287    6319    6405    6450    6467    6493    6519    6554
          6640     6680    6700    6727    6750    6826    6866    6969    7001    7099    7161    7217    7281    7328    7427
          7475     7592    7645    7751    7799    7904    7945    8039    8085    8177    8223    8325    8373    8487    8534
          8640     8692    8817    8863    9044    9095    9213    9264    9386    9432    9551    9606    9740    9791    9903
          9958    10142   10204   10365   10415   10520   10565   10680   10731   10784   10847   10891   11019   11067   11187
         11240    11376   11427   11437   11584   11633   11799   11844   12035   12090   12166   12240   12378   12424   12491
         12625    12679
BGNSET       1#    1416#
BGNSFT       1#    1416#
BGNSRV       1#    1416#   4942    4961    4987
BGNSUB       1#    1416#   5022    5052    5125    5155    5380    5407    5689    5712    5734    6222    6273    6399    6513
BGNSW        1#    1416#
BGNTST       1#    1416#   5013    5116    5268    5316    5375    5580    5624    5687    5800    6220    6396    6635    6819
          6963     7092    7274    7420    7585    7744    7897    8032    8170    8318    8480    8633    8810    9037    9206
          9379     9544    9733    9896   10135   10358   10513   10673   10840   11012   11180   11369   11577   11792   12028
         12371    12618   12785
BNCOMP       1#    1416#   4762    4776    4781    4825
BNERRO       1#    1416#
BREAK        1#    1416#   4276    4461
BRESET       1#    1416#
CKLOOP       1#    1416#   5038    5141    5287    5337
CLOCK        1#    1416#   4789
CLOSE        1#    1416#
CLRVEC       1#    1416#   5040    5096    5143    5250    5289    5297    5339    5351    7042
COMMEN       1#    1416#
DELAY        1#    1416#
DESCRI       1#    1416#   1969
DEVTYP       1#    1416#   1961
DISPAT       1#    1416#   1522
DISPLA       1#    1416#
DOCLN        1#    1416#   4841    5046    5149    5295    5345
DODU         1#    1416#   5043    5146    5292    5342
DORPT        1#    1416#
ENDAU        1#    1416#   4924
ENDAUT       1#    1416#   4861
ENDCLN       1#    1416#   4903
ENDCOM       1#    1416#
ENDDU        1#    1416#   4913
```

```
ENDHRD       1#    1416#   13151
ENDHW        1#    1416#    1589
ENDINI       1#    1416#    4844
ENDMOD       1#    1416#   13178
ENDMSG       1#    1416#    2923     2939     2960     2973     2986     3003     3022     3039     3052     3073     3099     3112     3124
          3137     3149     3174     3187     3207     3221     3236     3264     3292     3306     3335     3362     3389     3424     3436
          3474     3522     3541     3561     3581     3609     3628     3669     3696     3709     3729     3756     3768     3780     3792
          3804     3831     3851     3893     3905     3944     3968     3980
ENDPRO       1#    1416#    4739
ENDPTA       1#    1416#
ENDRPT       1#    1416#    4720
ENDSEG       1#    1416#    5073     5090     5176     5193     5210     5227     5244     5348     5429     5453     5484     5508     5533
          5552     5600     5640     5656     5773     6250     6267     6316     6333     6447     6464     6490     6507     6551     6565
          6677     6697     6724     6744     6779     6857     6931     6995     7039     7144     7211     7238     7319     7378     7465
          7557     7630     7695     7789     7851     7935     8001     8077     8138     8215     8276     8363     8439     8525     8600
          8678     8757     8855     8983     9082     9155     9251     9325     9424     9489     9589     9662     9778     9843     9941
         10005    10180    10289    10403    10471    10558    10624    10718    10778    10797    10885    10956    11057    11132    11225
         11312    11414    11500    11513    11622    11718    11837    11926    12073    12160    12234    12310    12416    12484    12555
         12663    12761
ENDSET       1#    1416#
ENDSFT       1#    1416#
ENDSRV       1#    1416#    4945     4965     4993
ENDSUB       1#    1416#    5049     5093     5152     5247     5403     5555     5709     5731     5776     6270     6353     6510     6584
ENDSW        1#    1416#
ENDTST       1#    1416#    5099     5253     5301     5354     5558     5605     5659     5779     5864     6356     6595     6788     6934
          7054     7245     7381     7560     7713     7872     8004     8141     8279     8442     8603     8762     8986     9178     9348
          9492     9687     9870    10031    10292    10481    10627    10806    10959    11140    11321    11523    11743    11940    12313
         12564    12764    13013
EQUALS       1#    1416#    1602
ERRDF        1#    1416#    5033     5136     5282     5332
ERRHRD       1#    1416#    5067     5084     5170     5187     5204     5221     5238     5394     5423     5447     5478     5502     5527
          5546     5594     5634     5650     5703     5725     5750     5767     5804     5821     5845     5858     6241     6261     6307
          6327     6347     6438     6458     6481     6501     6542     6559     6576     6668     6688     6715     6735     6763     6773
          6842     6850     6870     6891     6898     6918     6925     6989     7017     7033     7048     7119     7129     7165     7193
          7205     7232     7300     7310     7332     7344     7365     7372     7446     7456     7479     7500     7509     7532     7539
          7551     7611     7621     7649     7664     7673     7690     7703     7770     7780     7803     7817     7826     7845     7859
          7920     7928     7950     7964     7973     7990     8058     8068     8089     8103     8110     8119     8132     8196     8206
          8227     8241     8248     8257     8270     8344     8354     8377     8396     8405     8423     8433     8506     8516     8538
          8557     8566     8584     8594     8659     8669     8700     8714     8723     8741     8751     8836     8846     8867     8885
          8894     8915     8928     8937     8949     8958     8967     8977     9063     9073     9103     9117     9126     9149     9163
          9232     9242     9272     9286     9295     9319     9333     9405     9415     9436     9451     9460     9473     9483     9570
          9580     9613     9627     9636     9656     9670     9759     9769     9799     9813     9822     9837     9851     9922     9932
          9962     9976     9985     9999    10013    10161    10171    10213    10227    10236    10253    10269    10283    10384    10394
         10423    10446    10454    10465    10539    10549    10569    10585    10594    10608    10618    10699    10709    10735    10750
         10759    10772    10788    10866    10876    10895    10917    10926    10940    10950    11038    11048    11074    11093    11102
         11115    11125    11206    11216    11244    11263    11272    11288    11296    11306    11395    11405    11447    11461    11470
         11484    11494    11603    11613    11646    11660    11669    11686    11701    11712    11726    11818    11828    11857    11871
         11880    11894    11909    11920    11934    12054    12064    12098    12112    12121    12143    12154    12175    12189    12195
         12215    12226    12247    12261    12270    12294    12304    12397    12407    12431    12445    12454    12468    12478    12498
         12512    12521    12536    12549    12644    12654    12683    12697    12706    12719    12750    12811    12824    12845    12858
         12889    12902    12934    12946
ERROR        1#    1416#
ERRSF        1#    1416#
ERRSOF       1#    1416#
ERRTBL       1#    1416#
ESCAPE       1#    1416#    5399     5755     5809     5850     6246     6312     6443     6486     6547     6673     6693     6720     6740
```

```
          6875   7022   7108   7124   7134   7170   7290   7305   7315   7337   7349   7436   7451   7461   7484
          7505   7514   7601   7616   7626   7654   7669   7678   7708   7760   7775   7785   7808   7822   7831
          7864   7955   7969   8048   8063   8073   8094   8115   8124   8186   8201   8211   8232   8253   8262
          8334   8349   8359   8382   8401   8410   8496   8511   8521   8543   8562   8571   8649   8664   8674
          8705   8719   8728   8826   8841   8851   8872   8890   8899   9053   9068   9078   9108   9122   9131
          9168   9222   9237   9247   9277   9291   9300   9338   9395   9410   9420   9441   9456   9465   9560
          9575   9585   9618   9632   9641   9675   9749   9764   9774   9804   9818   9827   9856   9912   9927
          9937   9967   9981   9990  10018  10151  10166  10176  10218  10232  10241  10374  10389  10399  10428
         10451  10529  10544  10554  10574  10590  10599  10689  10704  10714  10740  10755  10764  10793  10856
         10871  10881  10900  10922  10931  11028  11043  11053  11079  11098  11107  11196  11211  11221  11249
         11268  11277  11385  11400  11410  11452  11466  11475  11593  11608  11618  11651  11665  11674  11731
         11808  11823  11833  11862  11876  11885  12044  12059  12069  12103  12117  12126  12180  12200  12252
         12266  12275  12387  12402  12412  12436  12450  12459  12503  12517  12526  12541  12634  12649  12659
         12688  12702  12711  12724  12755  12795  12816  12829  12850  12863  12894  12907  12939  12951
EXIT        1#  1416#   4717   4890   4910   4921  12789
FEQUAL      1#  1416#
GETBYT      1#  1416#
GETPRI      1#  1416#
GETWOR      1#  1416#
GMANIA      1#  1416#
GMANID      1#  1416#
GMANIL      1#  1416#
GPHARD      1#  1416#   4821
GPRMA       1#  1416#  13141  13146
GPRMD       1#  1416#
GPRML       1#  1416#
HEADER      1#  1416#   1428
INLOOP      1#  1416#
IOSETU      1#  1416#
IOSTAR      1#  1416#
KT11        1#  1416#
LASTAD      1#  1416#
MANUAL      1#  1416#
MEMORY      1#  1416#   4784
MSBYTE      1#  1416#  1429#   1435   1436   1437
MSCHEC      1#  1416#  4718#  4891#  4911#  4922# 12790#
MSCNTO      1#  1416# 13142# 13147#
MSCOUN      1#  1416#  2917#  2930#  2946#  2953#  2967#  2980#  2993#  3010#  3016#  3029#  3046#  3059#  3065#
         3080#  3086#  3093#  3106#  3119#  3131#  3144#  3156#  3162#  3168#  3181#  3194#  3201#  3214#  3228#
         3243#  3250#  3257#  3271#  3278#  3285#  3299#  3313#  3320#  3328#  3342#  3348#  3355#  3369#  3375#
         3382#  3397#  3405#  3415#  3431#  3443#  3449#  3458#  3465#  3483#  3490#  3496#  3502#  3509#  3516#
         3529#  3535#  3548#  3554#  3568#  3574#  3588#  3596#  3603#  3616#  3622#  3635#  3641#  3648#  3655#
         3662#  3676#  3682#  3690#  3703#  3716#  3723#  3736#  3743#  3749#  3763#  3775#  3787#  3799#  3811#
         3817#  3824#  3838#  3844#  3858#  3865#  3872#  3879#  3886#  3900#  3914#  3922#  3930#  3938#  3951#
         3957#  3963#  3975#  4386#  4422#  4539#  4554#  4575#  4641#  4656#  4669#  4684#  4697#  4796#  4882#
        12977# 12984# 12991# 12998# 13005#
MSDATA      1#  1416#  1429#   1438   1440   1442   1444   1446   1448   1450   1452   1454   1456   1458   1460
          1462   1464   1466  1468#   1470   1472   1475   1478   1480   1482   1484   1486   1488   1490   1492
          1494   1496   1498   1500   1502   1504   1506   1508   1510   1512  1962#  1970#
MSDECR      1#  1416#  1590#  2924#  2940#  2961#  2974#  2987#  3004#  3023#  3040#  3053#  3074#  3100#  3113#
         3125#  3138#  3150#  3175#  3188#  3208#  3222#  3237#  3265#  3293#  3307#  3336#  3363#  3390#  3425#
         3437#  3475#  3523#  3542#  3562#  3582#  3610#  3629#  3670#  3697#  3710#  3730#  3757#  3769#  3781#
         3793#  3805#  3832#  3852#  3894#  3906#  3945#  3969#  3981#  4721#  4740#  4845#  4862#  4904#  4914#
         4925#  4946#  4966#  4994#  5050#  5074#  5091#  5094#  5100#  5153#  5177#  5194#  5211#  5228#  5245#
         5248#  5254#  5302#  5349#  5355#  5404#  5430#  5454#  5485#  5509#  5534#  5553#  5556#  5559#  5601#
         5606#  5641#  5657#  5660#  5710#  5732#  5774#  5777#  5780#  5865#  6251#  6268#  6271#  6317#  6334#
```

```
           6354#   6357#   6448#   6465#   6491#   6508#   6511#   6552#   6566#   6585#   6596#   6678#   6698#   6725#   6745#
           6780#   6789#   6858#   6932#   6935#   6996#   7040#   7055#   7145#   7212#   7239#   7246#   7320#   7379#   7382#
           7466#   7558#   7561#   7631#   7696#   7714#   7790#   7852#   7873#   7936#   8002#   8005#   8078#   8139#   8142#
           8216#   8277#   8280#   8364#   8440#   8443#   8526#   8601#   8604#   8679#   8758#   8763#   8856#   8984#   8987#
           9083#   9156#   9179#   9252#   9326#   9349#   9425#   9490#   9493#   9590#   9663#   9688#   9779#   9844#   9871#
           9942#  10006#  10032#  10181#  10290#  10293#  10404#  10472#  10482#  10559#  10625#  10628#  10719#  10779#  10798#
          10807#  10886#  10957#  10960#  11058#  11133#  11141#  11226#  11313#  11322#  11415#  11501#  11514#  11524#  11623#
          11719#  11744#  11838#  11927#  11941#  12074#  12161#  12235#  12311#  12314#  12417#  12485#  12556#  12565#  12664#
          12762#  12765#  13014#  13152#  13179#
MSDEFA        1#   1416#  13142#  13147#
MSENDE        1#   1416#   1590#   2924#   2940#   2961#   2974#   2987#   3004#   3023#   3040#   3053#   3074#   3100#   3113#
           3125#   3138#   3150#   3175#   3188#   3208#   3222#   3237#   3265#   3293#   3307#   3336#   3363#   3390#   3425#
           3437#   3475#   3523#   3542#   3562#   3582#   3610#   3629#   3670#   3697#   3710#   3730#   3757#   3769#   3781#
           3793#   3805#   3832#   3852#   3894#   3906#   3945#   3969#   3981#   4721#   4845#   4862#   4904#   4914#   4925#
           4946#   4966#   4994#   5050#   5074#   5091#   5094#   5100#   5153#   5177#   5194#   5211#   5228#   5245#   5248#
           5254#   5302#   5349#   5355#   5404#   5430#   5454#   5485#   5509#   5534#   5553#   5556#   5559#   5601#   5606#
           5641#   5657#   5660#   5710#   5732#   5774#   5777#   5780#   5865#   6251#   6268#   6271#   6317#   6334#   6354#
           6357#   6448#   6465#   6491#   6508#   6511#   6552#   6566#   6585#   6596#   6678#   6698#   6725#   6745#   6780#
           6789#   6858#   6932#   6935#   6996#   7040#   7055#   7145#   7212#   7239#   7246#   7320#   7379#   7382#   7466#
           7558#   7561#   7631#   7696#   7714#   7790#   7852#   7873#   7936#   8002#   8005#   8078#   8139#   8142#   8216#
           8277#   8280#   8364#   8440#   8443#   8526#   8601#   8604#   8679#   8758#   8763#   8856#   8984#   8987#   9083#
           9156#   9179#   9252#   9326#   9349#   9425#   9490#   9493#   9590#   9663#   9688#   9779#   9844#   9871#   9942#
          10006#  10032#  10181#  10290#  10293#  10404#  10472#  10482#  10559#  10625#  10628#  10719#  10779#  10798#  10807#
          10886#  10957#  10960#  11058#  11133#  11141#  11226#  11313#  11322#  11415#  11501#  11514#  11524#  11623#  11719#
          11744#  11838#  11927#  11941#  12074#  12161#  12235#  12311#  12314#  12417#  12485#  12556#  12565#  12664#  12762#
          12765#  13014#  13152#  13179#
MSERRI        1#   1416#   5034#   5068#   5085#   5137#   5171#   5188#   5205#   5222#   5239#   5283#   5333#   5395#   5424#
           5448#   5479#   5503#   5528#   5547#   5595#   5635#   5651#   5704#   5726#   5751#   5768#   5805#   5822#   5846#
           5859#   6242#   6262#   6308#   6328#   6348#   6439#   6459#   6482#   6502#   6543#   6560#   6577#   6669#   6689#
           6716#   6736#   6764#   6774#   6843#   6851#   6871#   6892#   6899#   6919#   6926#   6990#   7018#   7034#   7049#
           7120#   7130#   7166#   7194#   7206#   7233#   7301#   7311#   7333#   7345#   7366#   7373#   7447#   7457#   7480#
           7501#   7510#   7533#   7540#   7552#   7612#   7622#   7650#   7665#   7674#   7691#   7704#   7771#   7781#   7804#
           7818#   7827#   7846#   7860#   7921#   7929#   7951#   7965#   7974#   7991#   8059#   8069#   8090#   8104#   8111#
           8120#   8133#   8197#   8207#   8228#   8242#   8249#   8258#   8271#   8345#   8355#   8378#   8397#   8406#   8424#
           8434#   8507#   8517#   8539#   8558#   8567#   8585#   8595#   8660#   8670#   8701#   8715#   8724#   8742#   8752#
           8837#   8847#   8868#   8886#   8895#   8916#   8929#   8938#   8950#   8959#   8968#   8978#   9064#   9074#   9104#
           9118#   9127#   9150#   9164#   9233#   9243#   9273#   9287#   9296#   9320#   9334#   9406#   9416#   9437#   9452#
           9461#   9474#   9484#   9571#   9581#   9614#   9628#   9637#   9657#   9671#   9760#   9770#   9800#   9814#   9823#
     [illegible]   9852#   9923#   9933#   9963#   9977#   9986#  10000#  10014#  10162#  10172#  10214#  10228#  10237#  10254#
          10270#  10284#  10385#  10395#  10424#  10447#  10455#  10466#  10540#  10550#  10570#  10586#  10595#  10609#  10619#
          10700#  10710#  10736#  10751#  10760#  10773#  10789#  10867#  10877#  10896#  10918#  10927#  10941#  10951#  11039#
          11049#  11075#  11094#  11103#  11116#  11126#  11207#  11217#  11245#  11264#  11273#  11289#  11297#  11307#  11396#
          11406#  11448#  11462#  11471#  11485#  11495#  11604#  11614#  11647#  11661#  11670#  11687#  11702#  11713#  11727#
          11819#  11829#  11858#  11872#  11881#  11895#  11910#  11921#  11935#  12055#  12065#  12099#  12113#  12122#  12144#
          12155#  12176#  12190#  12196#  12216#  12227#  12248#  12262#  12271#  12295#  12305#  12398#  12408#  12432#  12446#
          12455#  12469#  12479#  12499#  12513#  12522#  12537#  12550#  12645#  12655#  12684#  12698#  12707#  12720#  12751#
          12812#  12825#  12846#  12859#  12890#  12903#  12935#  12947#
MSESCA        1#   1416#   5400#   5401    5756#   5757    5810#   5811    5851#   5852    6247#   6248    6313#   6314    6444#
           6445    6487#   6488    6548#   6549    6674#   6675    6694#   6695    6721#   6722    6741#   6742    6876#   6877
           7023#   7024    7109#   7110    7125#   7126    7135#   7136    7171#   7172    7291#   7292    7306#   7307    7316#
           7317    7338#   7339    7350#   7351    7437#   7438    7452#   7453    7462#   7463    7485#   7486    7506#   7507
           7515#   7516    7602#   7603    7617#   7618    7627#   7628    7655#   7656    7670#   7671    7679#   7680    7709#
           7710    7761#   7762    7776#   7777    7786#   7787    7809#   7810    7823#   7824    7832#   7833    7865#   7866
           7956#   7957    7970#   7971    8049#   8050    8064#   8065    8074#   8075    8095#   8096    8116#   8117    8125#
           8126    8187#   8188    8202#   8203    8212#   8213    8233#   8234    8254#   8255    8263#   8264    8335#   8336
           8350#   8351    8360#   8361    8383#   8384    8402#   8403    8411#   8412    8497#   8498    8512#   8513    8522#
```

```
          8523   8544#  8545   8563#  8564   8572#  8573   8650#  8651   8665#  8666   8675#  8676   8706#  8707
          8720#  8721   8729#  8730   8827#  8828   8842#  8843   8852#  8853   8873#  8874   8891#  8892   8900#
          8901   9054#  9055   9069#  9070   9079#  9080   9109#  9110   9123#  9124   9132#  9133   9169#  9170
          9223#  9224   9238#  9239   9248#  9249   9278#  9279   9292#  9293   9301#  9302   9339#  9340   9396#
          9397   9411#  9412   9421#  9422   9442#  9443   9457#  9458   9466#  9467   9561#  9562   9576#  9577
          9586#  9587   9619#  9620   9633#  9634   9642#  9643   9676#  9677   9750#  9751   9765#  9766   9775#
          9776   9805#  9806   9819#  9820   9828#  9829   9857#  9858   9913#  9914   9928#  9929   9938#  9939
          9968#  9969   9982#  9983   9991#  9992   10019# 10020  10152# 10153  10167# 10168  10177# 10178  10219#
          10220  10233# 10234  10242# 10243  10375# 10376  10390# 10391  10400# 10401  10429# 10430  10452# 10453
          10530# 10531  10545# 10546  10555# 10556  10575# 10576  10591# 10592  10600# 10601  10690# 10691  10705#
          10706  10715# 10716  10741# 10742  10756# 10757  10765# 10766  10794# 10795  10857# 10858  10872# 10873
          10882# 10883  10901# 10902  10923# 10924  10932# 10933  11029# 11030  11044# 11045  11054# 11055  11080#
          11081  11099# 11100  11108# 11109  11197# 11198  11212# 11213  11222# 11223  11250# 11251  11269# 11270
          11278# 11279  11386# 11387  11401# 11402  11411# 11412  11453# 11454  11467# 11468  11476# 11477  11594#
          11595  11609# 11610  11619# 11620  11652# 11653  11666# 11667  11675# 11676  11732# 11733  11809# 11810
          11824# 11825  11834# 11835  11863# 11864  11877# 11878  11886# 11887  12045# 12046  12060# 12061  12070#
          12071  12104# 12105  12118# 12119  12127# 12128  12181# 12182  12201# 12202  12253# 12254  12267# 12268
          12276# 12277  12388# 12389  12403# 12404  12413# 12414  12437# 12438  12451# 12452  12460# 12461  12504#
          12505  12518# 12519  12527# 12528  12542# 12543  12635# 12636  12650# 12651  12660# 12661  12689# 12690
          12703# 12704  12712# 12713  12725# 12726  12756# 12757  12796# 12797  12817# 12818  12830# 12831  12851#
          12852  12864# 12865  12895# 12896  12908# 12909  12940# 12941  12952# 12953
MSESCS       1#  1416#  5400#  5756#  5810#  5851#  6247#  6313#  6444#  6487#  6548#  6674#  6694#  6721#  6741#
          6876#  7023#  7109#  7125#  7135#  7171#  7291#  7306#  7316#  7338#  7350#  7437#  7452#  7462#  7485#
          7506#  7515#  7602#  7617#  7627#  7655#  7670#  7679#  7709#  7761#  7776#  7786#  7809#  7823#  7832#
          7865#  7956#  7970#  8049#  8064#  8074#  8095#  8116#  8125#  8187#  8202#  8212#  8233#  8254#  8263#
          8335#  8350#  8360#  8383#  8402#  8411#  8497#  8512#  8522#  8544#  8563#  8572#  8650#  8665#  8675#
          8706#  8720#  8729#  8827#  8842#  8852#  8873#  8891#  8900#  9054#  9069#  9079#  9109#  9123#  9132#
          9169#  9223#  9238#  9248#  9278#  9292#  9301#  9339#  9396#  9411#  9421#  9442#  9457#  9466#  9561#
          9576#  9586#  9619#  9633#  9642#  9676#  9750#  9765#  9775#  9805#  9819#  9828#  9857#  9913#  9928#
          9938#  9968#  9982#  9991#  10019# 10152# 10167# 10177# 10219# 10233# 10242# 10375# 10390# 10400# 10429#
          10452# 10530# 10545# 10555# 10575# 10591# 10600# 10690# 10705# 10715# 10741# 10756# 10765# 10794# 10857#
          10872# 10882# 10901# 10923# 10932# 11029# 11044# 11054# 11080# 11099# 11108# 11197# 11212# 11222# 11250#
          11269# 11278# 11386# 11401# 11411# 11453# 11467# 11476# 11594# 11609# 11619# 11652# 11666# 11675# 11732#
          11809# 11824# 11834# 11863# 11877# 11886# 12045# 12060# 12070# 12104# 12118# 12127# 12181# 12201# 12253#
          12267# 12276# 12388# 12403# 12413# 12437# 12451# 12460# 12504# 12518# 12527# 12542# 12635# 12650# 12660#
          12689# 12703# 12712# 12725# 12756# 12796# 12817# 12830# 12851# 12864# 12895# 12908# 12940# 12952#
MSEXCP       1#  1416#  13142# 13147#
MSEXIT       1#  1416#  4718#  4891#  4892   4911#  4922#  12790# 12791
MSEXSE       1#  1416#  4718#  4891#  4911#  4922#  12790#
MSEXTJ       1#  1416#  4718#  4719   4891#  4911#  4912   4922#  4923   12790#
MSGEN        1#  1416#  1429#  1438#  1440#  1442#  1444#  1446#  1448#  1450#  1452#  1454#  1456#  1458#  1460#
          1462#  1464#  1466#  1468#  1470#  1472#  1475#  1478#  1480#  1482#  1484#  1486#  1488#  1490#  1492#
          1494#  1496#  1498#  1500#  1502#  1504#  1506#  1508#  1510#  1512#  1524#  1583#  1584#  1590#  1962#
          1970#  2915#  2924#  2928#  2940#  2944#  2961#  2965#  2974#  2978#  2987#  2991#  3004#  3008#  3023#
          3027#  3040#  3044#  3053#  3057#  3074#  3078#  3100#  3104#  3113#  3117#  3125#  3129#  3138#  3142#
          3150#  3154#  3175#  3179#  3188#  3192#  3208#  3212#  3222#  3226#  3237#  3241#  3265#  3269#  3293#
          3297#  3307#  3311#  3336#  3340#  3363#  3367#  3390#  3394#  3425#  3429#  3437#  3441#  3475#  3479#
          3523#  3527#  3542#  3546#  3562#  3566#  3582#  3586#  3610#  3614#  3629#  3633#  3670#  3674#  3697#
          3701#  3710#  3714#  3730#  3734#  3757#  3761#  3769#  3773#  3781#  3785#  3793#  3797#  3805#  3809#
          3832#  3836#  3852#  3856#  3894#  3898#  3906#  3910#  3945#  3949#  3969#  3973#  3981#  4716#  4721#
          4733#  4749#  4845#  4860#  4862#  4872#  4904#  4909#  4914#  4920#  4925#  4943#  4946#  4962#  4966#
          4988#  4994#  5014#  5023#  5050#  5053#  5074#  5091#  5094#  5100#  5117#  5126#  5153#  5156#  5177#
          5194#  5211#  5228#  5245#  5248#  5254#  5269#  5302#  5317#  5349#  5355#  5376#  5381#  5404#  5408#
          5430#  5454#  5485#  5509#  5534#  5553#  5556#  5559#  5581#  5601#  5606#  5625#  5641#  5657#  5660#
          5688#  5690#  5710#  5713#  5732#  5735#  5774#  5777#  5780#  5801#  5865#  6221#  6223#  6251#  6268#
          6271#  6274#  6317#  6334#  6354#  6357#  6397#  6400#  6448#  6465#  6491#  6508#  6511#  6514#  6552#
```

```
          6566#   6585#   6596#   6636#   6678#   6698#   6725#   6745#   6780#   6789#   6820#   6858#   6932#   6935#   6964#
          6996#   7040#   7055#   7093#   7145#   7212#   7239#   7246#   7275#   7320#   7379#   7382#   7421#   7466#   7558#
          7561#   7586#   7631#   7696#   7714#   7745#   7790#   7852#   7873#   7898#   7936#   8002#   8005#   8033#   8078#
          8139#   8142#   8171#   8216#   8277#   8280#   8319#   8364#   8440#   8443#   8481#   8526#   8601#   8604#   8634#
          8679#   8758#   8763#   8811#   8856#   8984#   8987#   9038#   9083#   9156#   9179#   9207#   9252#   9326#   9349#
          9380#   9425#   9490#   9493#   9545#   9590#   9663#   9688#   9734#   9779#   9844#   9871#   9897#   9942#  10006#
         10032#  10136#  10181#  10290#  10293#  10359#  10404#  10472#  10482#  10514#  10559#  10625#  10628#  10674#  10719#
         10779#  10798#  10807#  10841#  10886#  10957#  10960#  11013#  11058#  11133#  11141#  11181#  11226#  11313#  11322#
         11370#  11415#  11501#  11514#  11524#  11578#  11623#  11719#  11744#  11793#  11838#  11927#  11941#  12029#  12074#
         12161#  12235#  12311#  12314#  12372#  12417#  12485#  12556#  12565#  12619#  12664#  12762#  12765#  12786#  13014#
         13140#  13153#
MSGENB       1#   1416#
MSGETS       1#   1416#   1590#   2924#   2940#   2961#   2974#   2987#   3004#   3023#   3040#   3053#   3074#   3100#   3113#
          3125#   3138#   3150#   3175#   3188#   3208#   3222#   3237#   3265#   3293#   3307#   3336#   3363#   3390#   3425#
          3437#   3475#   3523#   3542#   3562#   3582#   3610#   3629#   3670#   3697#   3710#   3730#   3757#   3769#   3781#
          3793#   3805#   3832#   3852#   3894#   3906#   3945#   3969#   3981#   4721#   4740#   4845#   4862#   4904#   4914#
          4925#   4946#   4966#   4994#   5050#   5074#   5091#   5094#   5100#   5153#   5177#   5194#   5211#   5228#   5245#
          5248#   5254#   5302#   5349#   5355#   5404#   5430#   5454#   5485#   5509#   5534#   5553#   5556#   5559#   5601#
          5606#   5641#   5657#   5660#   5710#   5732#   5774#   5777#   5780#   5865#   6251#   6268#   6271#   6317#   6334#
          6354#   6357#   6448#   6465#   6491#   6508#   6511#   6552#   6566#   6585#   6596#   6678#   6698#   6725#   6745#
          6780#   6789#   6858#   6932#   6935#   6996#   7040#   7055#   7145#   7212#   7239#   7246#   7320#   7379#   7382#
          7466#   7558#   7561#   7631#   7696#   7714#   7790#   7852#   7873#   7936#   8002#   8005#   8078#   8139#   8142#
          8216#   8277#   8280#   8364#   8440#   8443#   8526#   8601#   8604#   8679#   8758#   8763#   8856#   8984#   8987#
          9083#   9156#   9179#   9252#   9326#   9349#   9425#   9490#   9493#   9590#   9663#   9688#   9779#   9844#   9871#
          9942#  10006#  10032#  10181#  10290#  10293#  10404#  10472#  10482#  10559#  10625#  10628#  10719#  10779#  10798#
         10807#  10886#  10957#  10960#  11058#  11133#  11141#  11226#  11313#  11322#  11415#  11501#  11514#  11524#  11623#
         11719#  11744#  11838#  11927#  11941#  12074#  12161#  12235#  12311#  12314#  12417#  12485#  12556#  12565#  12664#
         12762#  12765#  13014#  13152#  13179#
MSGETT       1#   1416#   4718#   4891#   4911#   4922#   5400#   5756#   5810#   5851#   6247#   6313#   6444#   6487#   6548#
          6674#   6694#   6721#   6741#   6876#   7023#   7109#   7125#   7135#   7171#   7291#   7306#   7316#   7338#   7350#
          7437#   7452#   7462#   7485#   7506#   7515#   7602#   7617#   7627#   7655#   7670#   7679#   7709#   7761#   7776#
          7786#   7809#   7823#   7832#   7865#   7956#   7970#   8049#   8064#   8074#   8095#   8116#   8125#   8187#   8202#
          8212#   8233#   8254#   8263#   8335#   8350#   8360#   8383#   8402#   8411#   8497#   8512#   8522#   8544#   8563#
          8572#   8650#   8665#   8675#   8706#   8720#   8729#   8827#   8842#   8852#   8873#   8891#   8900#   9054#   9069#
          9079#   9109#   9123#   9132#   9169#   9223#   9238#   9248#   9278#   9292#   9301#   9339#   9396#   9411#   9421#
          9442#   9457#   9466#   9561#   9576#   9586#   9619#   9633#   9642#   9676#   9750#   9765#   9775#   9805#   9819#
          9828#   9857#   9913#   9928#   9938#   9968#   9982#   9991#  10019#  10152#  10167#  10177#  10219#  10233#  10242#
         10375#  10390#  10400#  10429#  10452#  10530#  10545#  10555#  10575#  10591#  10600#  10690#  10705#  10715#  10741#
         10756#  10765#  10794#  10857#  10872#  10882#  10901#  10923#  10932#  11029#  11044#  11054#  11080#  11099#  11108#
         11197#  11212#  11222#  11250#  11269#  11278#  11386#  11401#  11411#  11453#  11467#  11476#  11594#  11609#  11619#
         11652#  11666#  11675#  11732#  11809#  11824#  11834#  11863#  11877#  11886#  12045#  12060#  12070#  12104#  12118#
         12127#  12181#  12201#  12253#  12267#  12276#  12388#  12403#  12413#  12437#  12451#  12460#  12504#  12518#  12527#
         12542#  12635#  12650#  12660#  12689#  12703#  12712#  12725#  12756#  12790#  12796#  12817#  12830#  12851#  12864#
         12895#  12908#  12940#  12952#
MSGNGB       1#   1416#   1420#   1429#   1438#   1440#   1442#   1444#   1446#   1448#   1450#   1452#   1454#   1456#   1458#
          1460#   1462#   1464#   1466#   1468#   1470#   1472#   1475#   1478#   1480#   1482#   1484#   1486#   1488#   1490#
          1492#   1494#   1496#   1498#   1500#   1502#   1504#   1506#   1508#   1510#   1512#   1523#   1524    1582#   1583
          1584    1962#   1970#   2915#   2928#   2944#   2965#   2978#   2991#   3008#   3027#   3044#   3057#   3078#   3104#
          3117#   3129#   3142#   3154#   3179#   3192#   3212#   3226#   3241#   3269#   3297#   3311#   3340#   3367#   3394#
          3429#   3441#   3479#   3527#   3546#   3566#   3586#   3614#   3633#   3674#   3701#   3714#   3734#   3761#   3773#
          3785#   3797#   3809#   3836#   3856#   3898#   3910#   3949#   3973#   4716#   4733#   4749#   4860#   4872#   4909#
          4920#   4943#   4962#   4988#  13139#  13140
MSGNIN       1#   1416#   1429#   1430    1431    1432    1433    1434    1435#   1436#   1437#   1438#   1439    1440#   1441
          1442#   1443    1444#   1445    1446#   1447    1448#   1449    1450#   1451    1452#   1453    1454#   1455    1456#
          1457    1458#   1459    1460#   1461    1462#   1463    1464#   1465    1466#   1467    1468#   1469    1470#   1471
          1472#   1473    1474    1475#   1476    1477#   1478#   1479    1480#   1481    1482#   1483    1484#   1485    1486#
```

1487	1488#	1489	1490#	1491	1492#	1493	1494#	1495	1496#	1497	1498#	1499	1500#	1501
1502#	1503	1504#	1505	1506#	1507	1508#	1509	1510#	1511	1512#	1513	1523#	1525#	1526#
1527#	1528#	1529#	1530#	1531#	1532#	1533#	1534#	1535#	1536#	1537#	1538#	1539#	1540#	1541#
1542#	1543#	1544#	1545#	1546#	1547#	1548#	1549#	1550#	1551#	1552#	1553#	1554#	1555#	1556#
1557#	1558#	1559#	1560#	1561#	1562#	1563#	1564#	1565#	1566#	1567#	1568#	1569#	1570#	1582#
1962#	1963	1964	1970#	1971	1975	2917#	2918#	2919#	2920	2921#	2922	2925#	2930#	2931#
2932#	2933#	2934#	2935	2936#	2937	2941#	2946#	2947#	2948#	2949	2950#	2951	2953#	2954#
2955#	2956#	2957	2958#	2959	2962#	2967#	2968#	2969	2970#	2971	2975#	2980#	2981#	2982
2983#	2984	2988#	2993#	2994#	2995#	2996#	2997#	2998#	2999	3000#	3001	3005#	3010#	3011#
3012	3013#	3014	3016#	3017#	3018#	3019	3020#	3021	3024#	3029#	3030#	3031#	3032#	3033#
3034#	3035	3036#	3037	3041#	3046#	3047#	3048	3049#	3050	3054#	3059#	3060#	3061	3062#
3063	3065#	3066#	3067#	3068#	3069#	3070	3071#	3072	3075#	3080#	3081#	3082	3083#	3084
3086#	3087#	3088#	3089	3090#	3091	3093#	3094#	3095#	3096	3097#	3098	3101#	3106#	3107#
3108	3109#	3110	3114#	3119#	3120#	3121	3122#	3123	3126#	3131#	3132#	3133#	3134	3135#
3136	3139#	3144#	3145#	3146	3147#	3148	3151#	3156#	3157#	3158	3159#	3160	3162#	3163#
3164	3165#	3166	3168#	3169#	3170#	3171	3172#	3173	3176#	3181#	3182#	3183#	3184	3185#
3186	3189#	3194#	3195#	3196#	3197	3198#	3199	3201#	3202#	3203#	3204	3205#	3206	3209#
3214#	3215#	3216#	3217#	3218	3219#	3220	3223#	3228#	3229#	3230#	3231#	3232#	3233	3234#
3235	3238#	3243#	3244#	3245#	3246	3247#	3248	3250#	3251#	3252#	3253	3254#	3255	3257#
3258#	3259#	3260#	3261	3262#	3263	3266#	3271#	3272#	3273#	3274	3275#	3276	3278#	3279#
3280#	3281	3282#	3283	3285#	3286#	3287#	3288#	3289	3290#	3291	3294#	3299#	3300#	3301#
3302#	3303	3304#	3305	3308#	3313#	3314#	3315#	3316	3317#	3318	3320#	3321#	3322#	3323#
3324	3325#	3326	3328#	3329#	3330#	3331#	3332	3333#	3334	3337#	3342#	3343#	3344	3345#
3346	3348#	3349#	3350#	3351	3352#	3353	3355#	3356#	3357#	3358#	3359	3360#	3361	3364#
3369#	3370#	3371	3372#	3373	3375#	3376#	3377#	3378	3379#	3380	3382#	3383#	3384#	3385#
3386	3387#	3388	3391#	3397#	3398#	3399#	3400	3401#	3402	3405#	3406#	3407#	3408#	3409#
3410	3411#	3412	3415#	3416#	3417#	3418#	3419#	3420	3421#	3422	3426#	3431#	3432#	3433
3434#	3435	3438#	3443#	3444#	3445	3446#	3447	3449#	3450#	3451	3452#	3453	3458#	3459#
3460#	3461	3462#	3463	3465#	3466#	3467#	3468	3469#	3470	3476#	3483#	3484#	3485	3486#
3487	3490#	3491#	3492	3493#	3494	3496#	3497#	3498	3499#	3500	3502#	3503#	3504#	3505
3506#	3507	3509#	3510#	3511#	3512	3513#	3514	3516#	3517#	3518#	3519	3520#	3521	3524#
3529#	3530#	3531	3532#	3533	3535#	3536#	3537#	3538	3539#	3540	3543#	3548#	3549#	3550
3551#	3552	3554#	3555#	3556#	3557#	3558	3559#	3560	3563#	3568#	3569#	3570	3571#	3572
3574#	3575#	3576#	3577#	3578	3579#	3580	3583#	3588#	3589#	3590#	3591#	3592	3593#	3594
3596#	3597#	3598#	3599	3600#	3601	3603#	3604#	3605#	3606	3607#	3608	3611#	3616#	3617#
3618	3619#	3620	3622#	3623#	3624#	3625	3626#	3627	3630#	3635#	3636#	3637	3638#	3639
3641#	3642#	3643#	3644	3645#	3646	3648#	3649#	3650#	3651	3652#	3653	3655#	3656#	3657#
3658	3659#	3660	3662#	3663#	3664#	3665#	3666	3667#	3668	3671#	3676#	3677#	3678	3679#
3680	3682#	3683#	3684#	3685#	3686	3687#	3688	3690#	3691#	3692#	3693	3694#	3695	3698#
3703#	3704#	3705#	3706	3707#	3708	3711#	3716#	3717#	3718#	3719	3720#	3721	3723#	3724#
3725#	3726	3727#	3728	3731#	3736#	3737#	3738#	3739	3740#	3741	3743#	3744#	3745	3746#
3747	3749#	3750#	3751#	3752#	3753	3754#	3755	3758#	3763#	3764#	3765	3766#	3767	3770#
3775#	3776#	3777	3778#	3779	3782#	3787#	3788#	3789	3790#	3791	3794#	3799#	3800#	3801
3802#	3803	3806#	3811#	3812#	3813	3814#	3815	3817#	3818#	3819#	3820	3821#	3822	3824#
3825#	3826#	3827#	3828	3829#	3830	3833#	3838#	3839#	3840	3841#	3842	3844#	3845#	3846#
3847#	3848	3849#	3850	3853#	3858#	3859#	3860#	3861	3862#	3863	3865#	3866#	3867#	3868
3869#	3870	3872#	3873#	3874#	3875	3876#	3877	3879#	3880#	3881#	3882	3883#	3884	3886#
3887#	3888#	3889#	3890	3891#	3892	3895#	3900#	3901#	3902	3903#	3904	3907#	3914#	3915#
3916	3917#	3918	3922#	3923#	3924	3925#	3926	3930#	3931#	3932	3933#	3934	3938#	3939#
3940	3961#	3942	3946#	3951#	3952#	3953	3954#	3955	3957#	3958#	3959	3960#	3961	3963#
3964#	3965	3966#	3967	3970#	3975#	3976#	3977	3978#	3979	3982#	4233#	4234#	4247#	4248#
4277#	4386#	4387#	4388#	4389	4390#	4391	4422#	4423#	4424#	4425#	4426	4427#	4428	4462#
4539#	4540#	4541#	4542#	4543#	4544#	4545	4546#	4547	4554#	4555#	4556	4557#	4558	4575#
4576#	4577#	4578	4579#	4580	4641#	4642#	4643#	4644	4645#	4646	4656#	4657#	4658#	4659
4660#	4661	4669#	4670#	4671#	4672	4673#	4674	4684#	4685#	4686#	4687	4688#	4689	4697#
4698#	4699#	4700	4701#	4702	4718#	4719#	4722#	4754#	4755#	4757#	4759#	4760#	4763#	4774#

4775#	4777#	4779#	4780#	4782#	4785#	4786#	4790#	4791#	4792#	4794#	4796#	4797#	4798	4799#
4800	4809#	4810#	4811#	4812#	4813#	4814	4822#	4823#	4824#	4826#	4842#	4846#	4863#	4882#
4883#	4884	4885#	4886	4891#	4892#	4905#	4911#	4912#	4915#	4922#	4923#	4926#	4946#	4947
4966#	4967	4994#	4995	5016#	5017#	5018#	5019#	5020#	5021	5024#	5034#	5035#	5036#	5037#
5039#	5041#	5042#	5044#	5045#	5047#	5051#	5054#	5059#	5068#	5069#	5070#	5071#	5075#	5077#
5085#	5086#	5087#	5088#	5092#	5095#	5097#	5098#	5101#	5119#	5120#	5121#	5122#	5123#	5124
5127#	5137#	5138#	5139#	5140#	5142#	5144#	5145#	5147#	5148#	5150#	5154#	5157#	5162#	5171#
5172#	5173#	5174#	5178#	5180#	5188#	5189#	5190#	5191#	5195#	5197#	5205#	5206#	5207#	5208#
5212#	5214#	5222#	5223#	5224#	5225#	5229#	5231#	5239#	5240#	5241#	5242#	5246#	5249#	5251#
5252#	5255#	5271#	5272#	5273#	5274#	5275#	5276	5283#	5284#	5285#	5286#	5288#	5290#	5291#
5293#	5294#	5296#	5298#	5299#	5303#	5319#	5320#	5321#	5322#	5323#	5324	5326#	5333#	5334#
5335#	5336#	5338#	5340#	5341#	5343#	5344#	5346#	5350#	5352#	5353#	5356#	5382#	5395#	5396#
5397#	5398#	5400#	5401#	5405#	5409#	5414#	5424#	5425#	5426#	5427#	5431#	5442#	5448#	5449#
5450#	5451#	5455#	5468#	5479#	5480#	5481#	5482#	5486#	5497#	5503#	5504#	5505#	5506#	5510#
5518#	5528#	5529#	5530#	5531#	5535#	5537#	5547#	5548#	5549#	5550#	5554#	5557#	5560#	5587#
5595#	5596#	5597#	5598#	5602#	5607#	5629#	5635#	5636#	5637#	5638#	5642#	5645#	5651#	5652#
5653#	5654#	5658#	5661#	5691#	5704#	5705#	5706#	5707#	5711#	5714#	5726#	5727#	5728#	5729#
5733#	5736#	5751#	5752#	5753#	5754#	5756#	5757#	5760#	5768#	5769#	5770#	5771#	5775#	5778#
5781#	5805#	5806#	5807#	5808#	5810#	5811#	5822#	5823#	5824#	5825#	5846#	5847#	5848#	5849#
5851#	5852#	5859#	5860#	5861#	5862#	5866#	6224#	6229#	6242#	6243#	6244#	6245#	6247#	6248#
6252#	6254#	6262#	6263#	6264#	6265#	6269#	6272#	6275#	6288#	6308#	6309#	6310#	6311#	6313#
6314#	6318#	6320#	6328#	6329#	6330#	6331#	6335#	6348#	6349#	6350#	6351#	6355#	6358#	6401#
6406#	6439#	6440#	6441#	6442#	6444#	6445#	6449#	6451#	6459#	6460#	6461#	6462#	6466#	6468#
6482#	6483#	6484#	6485#	6487#	6488#	6492#	6494#	6502#	6503#	6504#	6505#	6509#	6512#	6515#
6520#	6543#	6544#	6545#	6546#	6548#	6549#	6553#	6555#	6560#	6561#	6562#	6563#	6567#	6577#
6578#	6579#	6580#	6586#	6597#	6641#	6669#	6670#	6671#	6672#	6674#	6675#	6679#	6681#	6689#
6690#	6691#	6692#	6694#	6695#	6699#	6701#	6716#	6717#	6718#	6719#	6721#	6722#	6726#	6728#
6736#	6737#	6738#	6739#	6741#	6742#	6746#	6751#	6764#	6765#	6766#	6767#	6774#	6775#	6776#
6777#	6781#	6790#	6827#	6843#	6844#	6845#	6846#	6851#	6852#	6853#	6854#	6859#	6867#	6871#
6872#	6873#	6874#	6876#	6877#	6892#	6893#	6894#	6895#	6899#	6900#	6901#	6902#	6919#	6920#
6921#	6922#	6926#	6927#	6928#	6929#	6933#	6936#	6970#	6972#	6973#	6974#	6975#	6976#	6977
6990#	6991#	6992#	6993#	6997#	7002#	7008#	7009#	7018#	7019#	7020#	7021#	7023#	7024#	7034#
7035#	7036#	7037#	7041#	7043#	7044#	7049#	7050#	7051#	7052#	7056#	7100#	7109#	7110#	7120#
7121#	7122#	7123#	7125#	7126#	7130#	7131#	7132#	7133#	7135#	7136#	7138#	7139#	7140#	7141#
7142#	7143	7146#	7162#	7166#	7167#	7168#	7169#	7171#	7172#	7181#	7182#	7194#	7195#	7196#
7197#	7206#	7207#	7208#	7209#	7213#	7218#	7233#	7234#	7235#	7236#	7240#	7247#	7282#	7291#
7292#	7301#	7302#	7303#	7304#	7306#	7307#	7311#	7312#	7313#	7314#	7316#	7317#	7321#	7329#
7333#	7334#	7335#	7336#	7338#	7339#	7345#	7346#	7347#	7348#	7350#	7351#	7366#	7367#	7368#
7369#	7373#	7374#	7375#	7376#	7380#	7383#	7428#	7437#	7438#	7447#	7448#	7449#	7450#	7452#
7453#	7457#	7458#	7459#	7460#	7462#	7463#	7467#	7476#	7480#	7481#	7482#	7483#	7485#	7486#
7501#	7502#	7503#	7504#	7506#	7507#	7510#	7511#	7512#	7513#	7515#	7516#	7533#	7534#	7535#
7536#	7540#	7541#	7542#	7543#	7552#	7553#	7554#	7555#	7559#	7562#	7593#	7602#	7603#	7612#
7613#	7614#	7615#	7617#	7618#	7622#	7623#	7624#	7625#	7627#	7628#	7632#	7646#	7650#	7651#
7652#	7653#	7655#	7656#	7665#	7666#	7667#	7668#	7670#	7671#	7674#	7675#	7676#	7677#	7679#
7680#	7691#	7692#	7693#	7694#	7697#	7704#	7705#	7706#	7707#	7709#	7710#	7715#	7752#	7761#
7762#	7771#	7772#	7773#	7774#	7776#	7777#	7781#	7782#	7783#	7784#	7786#	7787#	7791#	7800#
7804#	7805#	7806#	7807#	7809#	7810#	7818#	7819#	7820#	7821#	7823#	7824#	7827#	7828#	7829#
7830#	7832#	7833#	7846#	7847#	7848#	7849#	7853#	7860#	7861#	7862#	7863#	7865#	7866#	7874#
7905#	7921#	7922#	7923#	7924#	7929#	7930#	7931#	7932#	7937#	7946#	7951#	7952#	7953#	7954#
7956#	7957#	7965#	7966#	7967#	7968#	7970#	7971#	7974#	7975#	7976#	7977#	7991#	7992#	7993#
7994#	8003#	8006#	8040#	8049#	8050#	8059#	8060#	8061#	8062#	8064#	8065#	8069#	8070#	8071#
8072#	8074#	8075#	8079#	8086#	8090#	8091#	8092#	8093#	8095#	8096#	8104#	8105#	8106#	8107#
8111#	8112#	8113#	8114#	8116#	8117#	8120#	8121#	8122#	8123#	8125#	8126#	8133#	8134#	8135#
8136#	8140#	8143#	8178#	8187#	8188#	8197#	8198#	8199#	8200#	8202#	8203#	8207#	8208#	8209#
8210#	8212#	8213#	8217#	8224#	8228#	8229#	8230#	8231#	8233#	8234#	8242#	8243#	8244#	8245#
8249#	8250#	8251#	8252#	8254#	8255#	8258#	8259#	8260#	8261#	8263#	8264#	8271#	8272#	8273#

8274#	8278#	8281#	8326#	8335#	8336#	8345#	8346#	8347#	8348#	8350#	8351#	8355#	8356#	8357#
8358#	8360#	8361#	8365#	8374#	8378#	8379#	8380#	8381#	8383#	8384#	8397#	8398#	8399#	8400#
8402#	8403#	8406#	8407#	8408#	8409#	8411#	8412#	8424#	8425#	8426#	8427#	8434#	8435#	8436#
8437#	8441#	8444#	8488#	8497#	8498#	8507#	8508#	8509#	8510#	8512#	8513#	8517#	8518#	8519#
8520#	8522#	8523#	8527#	8535#	8539#	8540#	8541#	8542#	8544#	8545#	8558#	8559#	8560#	8561#
8563#	8564#	8567#	8568#	8569#	8570#	8572#	8573#	8585#	8586#	8587#	8588#	8595#	8596#	8597#
8598#	8602#	8605#	8641#	8650#	8651#	8660#	8661#	8662#	8663#	8665#	8666#	8670#	8671#	8672#
8673#	8675#	8676#	8680#	8693#	8701#	8702#	8703#	8704#	8706#	8707#	8715#	8716#	8717#	8718#
8720#	8721#	8724#	8725#	8726#	8727#	8729#	8730#	8742#	8743#	8744#	8745#	8752#	8753#	8754#
8755#	8759#	8764#	8818#	8827#	8828#	8837#	8838#	8839#	8840#	8842#	8843#	8847#	8848#	8849#
8850#	8852#	8853#	8857#	8864#	8868#	8869#	8870#	8871#	8873#	8874#	8886#	8887#	8888#	8889#
8891#	8892#	8895#	8896#	8897#	8898#	8900#	8901#	8916#	8917#	8918#	8919#	8929#	8930#	8931#
8932#	8938#	8939#	8940#	8941#	8950#	8951#	8952#	8953#	8959#	8960#	8961#	8962#	8968#	8969#
8970#	8971#	8978#	8979#	8980#	8981#	8985#	8988#	9045#	9054#	9055#	9064#	9065#	9066#	9067#
9069#	9070#	9074#	9075#	9076#	9077#	9079#	9080#	9084#	9096#	9104#	9105#	9106#	9107#	9109#
9110#	9118#	9119#	9120#	9121#	9123#	9124#	9127#	9128#	9129#	9130#	9132#	9133#	9150#	9151#
9152#	9153#	9157#	9164#	9165#	9166#	9167#	9169#	9170#	9180#	9214#	9223#	9224#	9233#	9234#
9235#	9236#	9238#	9239#	9243#	9244#	9245#	9246#	9248#	9249#	9253#	9265#	9273#	9274#	9275#
9276#	9278#	9279#	9287#	9288#	9289#	9290#	9292#	9293#	9296#	9297#	9298#	9299#	9301#	9302#
9320#	9321#	9322#	9323#	9327#	9334#	9335#	9336#	9337#	9339#	9340#	9350#	9387#	9396#	9397#
9406#	9407#	9408#	9409#	9411#	9412#	9416#	9417#	9418#	9419#	9421#	9422#	9426#	9433#	9437#
9438#	9439#	9440#	9442#	9443#	9452#	9453#	9454#	9455#	9457#	9458#	9461#	9462#	9463#	9464#
9466#	9467#	9474#	9475#	9476#	9477#	9484#	9485#	9486#	9487#	9491#	9494#	9552#	9561#	9562#
9571#	9572#	9573#	9574#	9576#	9577#	9581#	9582#	9583#	9584#	9586#	9587#	9591#	9607#	9614#
9615#	9616#	9617#	9619#	9620#	9628#	9629#	9630#	9631#	9633#	9634#	9637#	9638#	9639#	9640#
9642#	9643#	9657#	9658#	9659#	9660#	9664#	9671#	9672#	9673#	9674#	9676#	9677#	9689#	9741#
9750#	9751#	9760#	9761#	9762#	9763#	9765#	9766#	9770#	9771#	9772#	9773#	9775#	9776#	9780#
9792#	9800#	9801#	9802#	9803#	9805#	9806#	9814#	9815#	9816#	9817#	9819#	9820#	9823#	9824#
9825#	9826#	9828#	9829#	9838#	9839#	9840#	9841#	9845#	9852#	9853#	9854#	9855#	9857#	9858#
9872#	9904#	9913#	9914#	9923#	9924#	9925#	9926#	9928#	9929#	9933#	9934#	9935#	9936#	9938#
9939#	9943#	9959#	9963#	9964#	9965#	9966#	9968#	9969#	9977#	9978#	9979#	9980#	9982#	9983#
9986#	9987#	9988#	9989#	9991#	9992#	10000#	10001#	10002#	10003#	10007#	10014#	10015#	10016#	10017#
10019#	10020#	10033#	10143#	10152#	10153#	10162#	10163#	10164#	10165#	10167#	10168#	10172#	10173#	10174#
10175#	10177#	10178#	10182#	10205#	10214#	10215#	10216#	10217#	10219#	10220#	10228#	10229#	10230#	10231#
10233#	10234#	10237#	10238#	10239#	10240#	10242#	10243#	10254#	10255#	10256#	10257#	10270#	10271#	10272#
10273#	10284#	10285#	10286#	10287#	10291#	10294#	10366#	10375#	10376#	10385#	10386#	10387#	10388#	10390#
10391#	10395#	10396#	10397#	10398#	10400#	10401#	10405#	10416#	10424#	10425#	10426#	10427#	10429#	10430#
10447#	10448#	10449#	10450#	10452#	10453#	10455#	10456#	10457#	10458#	10466#	10467#	10468#	10469#	10473#
10483#	10521#	10530#	10531#	10540#	10541#	10542#	10543#	10545#	10546#	10550#	10551#	10552#	10553#	10555#
10556#	10560#	10566#	10570#	10571#	10572#	10573#	10575#	10576#	10586#	10587#	10588#	10589#	10591#	10592#
10595#	10596#	10597#	10598#	10600#	10601#	10609#	10610#	10611#	10612#	10619#	10620#	10621#	10622#	10626#
10629#	10681#	10690#	10691#	10700#	10701#	10702#	10703#	10705#	10706#	10710#	10711#	10712#	10713#	10715#
10716#	10720#	10732#	10736#	10737#	10738#	10739#	10741#	10742#	10751#	10752#	10753#	10754#	10756#	10757#
10760#	10761#	10762#	10763#	10765#	10766#	10773#	10774#	10775#	10776#	10780#	10785#	10789#	10790#	10791#
10792#	10794#	10795#	10799#	10808#	10848#	10857#	10858#	10867#	10868#	10869#	10870#	10872#	10873#	10877#
10878#	10879#	10880#	10882#	10883#	10887#	10892#	10896#	10897#	10898#	10899#	10901#	10902#	10918#	10919#
10920#	10921#	10923#	10924#	10927#	10928#	10929#	10930#	10932#	10933#	10941#	10942#	10943#	10944#	10951#
10952#	10953#	10954#	10958#	10961#	11020#	11029#	11030#	11039#	11040#	11041#	11042#	11044#	11045#	11049#
11050#	11051#	11052#	11054#	11055#	11059#	11068#	11075#	11076#	11077#	11078#	11080#	11081#	11094#	11095#
11096#	11097#	11099#	11100#	11103#	11104#	11105#	11106#	11108#	11109#	11116#	11117#	11118#	11119#	11126#
11127#	11128#	11129#	11134#	11142#	11188#	11197#	11198#	11207#	11208#	11209#	11210#	11212#	11213#	11217#
11218#	11219#	11220#	11222#	11223#	11227#	11241#	11245#	11246#	11247#	11248#	11250#	11251#	11264#	11265#
11266#	11267#	11269#	11270#	11273#	11274#	11275#	11276#	11278#	11279#	11289#	11290#	11291#	11292#	11297#
11298#	11299#	11300#	11307#	11308#	11309#	11310#	11314#	11323#	11377#	11386#	11387#	11396#	11397#	11398#
11399#	11401#	11402#	11406#	11407#	11408#	11409#	11411#	11412#	11416#	11428#	11438#	11448#	11449#	11450#
11451#	11453#	11454#	11462#	11463#	11464#	11465#	11467#	11468#	11471#	11472#	11473#	11474#	11476#	11477#

```
         11485#  11486#  11487#  11488#  11495#  11496#  11497#  11498#  11502#  11515#  11525#  11585#  11594#  11595#  11604#
         11605#  11606#  11607#  11609#  11610#  11614#  11615#  11616#  11617#  11619#  11620#  11624#  11634#  11647#  11648#
         11649#  11650#  11652#  11653#  11661#  11662#  11663#  11664#  11666#  11667#  11670#  11671#  11672#  11673#  11675#
         11676#  11687#  11688#  11689#  11690#  11702#  11703#  11704#  11705#  11713#  11714#  11715#  11716#  11720#  11727#
         11728#  11729#  11730#  11732#  11733#  11745#  11800#  11809#  11810#  11819#  11820#  11821#  11822#  11824#  11825#
         11829#  11830#  11831#  11832#  11834#  11835#  11839#  11845#  11858#  11859#  11860#  11861#  11863#  11864#  11872#
         11873#  11874#  11875#  11877#  11878#  11881#  11882#  11883#  11884#  11886#  11887#  11895#  11896#  11897#  11898#
         11910#  11911#  11912#  11913#  11921#  11922#  11923#  11924#  11928#  11935#  11936#  11937#  11938#  11942#  12036#
         12045#  12046#  12055#  12056#  12057#  12058#  12060#  12061#  12065#  12066#  12067#  12068#  12070#  12071#  12075#
         12091#  12099#  12100#  12101#  12102#  12104#  12105#  12113#  12114#  12115#  12116#  12118#  12119#  12122#  12123#
         12124#  12125#  12127#  12128#  12144#  12145#  12146#  12147#  12155#  12156#  12157#  12158#  12162#  12167#  12176#
         12177#  12178#  12179#  12181#  12182#  12190#  12191#  12192#  12193#  12196#  12197#  12198#  12199#  12201#  12202#
         12216#  12217#  12218#  12219#  12227#  12228#  12229#  12230#  12236#  12241#  12248#  12249#  12250#  12251#  12253#
         12254#  12262#  12263#  12264#  12265#  12267#  12268#  12271#  12272#  12273#  12274#  12276#  12277#  12295#  12296#
         12297#  12298#  12305#  12306#  12307#  12308#  12312#  12315#  12379#  12388#  12389#  12398#  12399#  12400#  12401#
         12403#  12404#  12408#  12409#  12410#  12411#  12413#  12414#  12418#  12425#  12432#  12433#  12434#  12435#  12437#
         12438#  12446#  12447#  12448#  12449#  12451#  12452#  12455#  12456#  12457#  12458#  12460#  12461#  12469#  12470#
         12471#  12472#  12479#  12480#  12481#  12482#  12486#  12492#  12499#  12500#  12501#  12502#  12504#  12505#  12513#
         12514#  12515#  12516#  12518#  12519#  12522#  12523#  12524#  12525#  12527#  12528#  12537#  12538#  12539#  12540#
         12542#  12543#  12550#  12551#  12552#  12553#  12557#  12566#  12626#  12635#  12636#  12645#  12646#  12647#  12648#
         12650#  12651#  12655#  12656#  12657#  12658#  12660#  12661#  12665#  12680#  12684#  12685#  12686#  12687#  12689#
         12690#  12698#  12699#  12700#  12701#  12703#  12704#  12707#  12708#  12709#  12710#  12712#  12713#  12720#  12721#
         12722#  12723#  12725#  12726#  12751#  12752#  12753#  12754#  12756#  12757#  12763#  12766#  12790#  12791#  12796#
         12797#  12812#  12813#  12814#  12815#  12817#  12818#  12825#  12826#  12827#  12828#  12830#  12831#  12846#  12847#
         12848#  12849#  12851#  12852#  12859#  12860#  12861#  12862#  12864#  12865#  12890#  12891#  12892#  12893#  12895#
         12896#  12903#  12904#  12905#  12906#  12908#  12909#  12935#  12936#  12937#  12938#  12940#  12941#  12947#  12948#
         12949#  12950#  12952#  12953#  12977#  12978#  12979#  12980   12981#  12982   12984#  12985#  12986#  12987   12988#
         12989   12991#  12992#  12993#  12994   12995#  12996   12998#  12999#  13000#  13001   13002#  13003   13005#  13006#
         13007#  13008   13009#  13010   13015#  13139#  13142#  13143   13144   13145   13147#  13148   13149   13150   13152#
MSGNLS       1#   1416#   5074#   5091#   5177#   5194#   5211#   5228#   5245#   5349#   5430#   5454#   5485#   5509#   5534#
          5553#   5601#   5641#   5657#   5774#   6251#   6268#   6317#   6334#   6448#   6465#   6491#   6508#   6552#   6566#
          6678#   6698#   6725#   6745#   6780#   6858#   6932#   6996#   7040#   7145#   7212#   7239#   7320#   7379#   7466#
          7558#   7631#   7696#   7790#   7852#   7936#   8002#   8078#   8139#   8216#   8277#   8364#   8440#   8526#   8601#
          8679#   8758#   8856#   8984#   9083#   9156#   9252#   9326#   9425#   9490#   9590#   9663#   9779#   9844#   9942#
         10006#  10181#  10290#  10404#  10472#  10559#  10625#  10719#  10779#  10798#  10886#  10957#  11058#  11133#  11226#
         11313#  11415#  11501#  11514#  11623#  11719#  11838#  11927#  12074#  12161#  12235#  12311#  12417#  12485#  12556#
         12664#  12762#
MSGNSU       1#   1416#   5023#   5053#   5126#   5156#   5381#   5408#   5690#   5713#   5735#   6223#   6274#   6400#   6514#
MSGNTA       1#   1416#   1590#   2924#   2940#   2961#   2974#   2987#   3004#   3023#   3040#   3053#   3074#   3100#   3113#
          3125#   3138#   3150#   3175#   3188#   3208#   3222#   3237#   3265#   3293#   3307#   3336#   3363#   3390#   3425#
          3437#   3475#   3523#   3542#   3562#   3582#   3610#   3629#   3670#   3697#   3710#   3730#   3757#   3769#   3781#
          3793#   3805#   3832#   3852#   3894#   3906#   3945#   3969#   3981#   4721#   4845#   4862#   4904#   4914#   4925#
          4946#   4966#   4994#   5050#   5094#   5100#   5153#   5248#   5254#   5302#   5355#   5404#   5556#   5559#   5606#
          5660#   5710#   5732#   5777#   5780#   5865#   6271#   6354#   6357#   6511#   6585#   6596#   6789#   6935#   7055#
          7246#   7382#   7561#   7714#   7873#   8005#   8142#   8280#   8443#   8604#   8763#   8987#   9179#   9349#   9493#
          9688#   9871#  10032#  10293#  10482#  10628#  10807#  10960#  11141#  11322#  11524#  11744#  11941#  12314#  12565#
         12765#  13014#  13152#  13153
MSGNTE       1#   1416#   5014#   5117#   5269#   5317#   5376#   5581#   5625#   5688#   5801#   6221#   6397#   6636#   6820#
          6964#   7093#   7275#   7421#   7586#   7745#   7898#   8033#   8171#   8319#   8481#   8634#   8811#   9038#   9207#
          9380#   9545#   9734#   9897#  10136#  10359#  10514#  10674#  10841#  11013#  11181#  11370#  11578#  11793#  12029#
         12372#  12619#  12786#
MSHAPT       1#   1416#   1429#
MSHNAP       1#   1416#   1429#   1468
MSINCR       1#   1416#   1420#   1582#   2915#   2921#   2925#   2928#   2936#   2941#   2944#   2950#   2958#   2962#   2965#
          2970#   2975#   2978#   2983#   2988#   2991#   3000#   3005#   3008#   3013#   3020#   3024#   3027#   3036#   3041#
          3044#   3049#   3054#   3057#   3062#   3071#   3075#   3078#   3083#   3090#   3097#   3101#   3104#   3109#   3114#
```

3117#	3122#	3126#	3129#	3135#	3139#	3142#	3147#	3151#	3154#	3159#	3165#	3172#	3176#	3179#
3185#	3189#	3192#	3198#	3205#	3209#	3212#	3219#	3223#	3226#	3234#	3238#	3241#	3247#	3254#
3262#	3266#	3269#	3275#	3282#	3290#	3294#	3297#	3304#	3308#	3311#	3317#	3325#	3333#	3337#
3340#	3345#	3352#	3360#	3364#	3367#	3372#	3379#	3387#	3391#	3394#	3401#	3411#	3421#	3426#
3429#	3434#	3438#	3441#	3446#	3452#	3462#	3469#	3476#	3479#	3486#	3493#	3499#	3506#	3513#
3520#	3524#	3527#	3532#	3539#	3543#	3546#	3551#	3559#	3563#	3566#	3571#	3579#	3583#	3586#
3593#	3600#	3607#	3611#	3614#	3619#	3626#	3630#	3633#	3638#	3645#	3652#	3659#	3667#	3671#
3674#	3679#	3687#	3694#	3698#	3701#	3707#	3711#	3714#	3720#	3727#	3731#	3734#	3740#	3746#
3754#	3758#	3761#	3766#	3770#	3773#	3778#	3782#	3785#	3790#	3794#	3797#	3802#	3806#	3809#
3814#	3821#	3829#	3833#	3836#	3841#	3849#	3853#	3856#	3862#	3869#	3876#	3883#	3891#	3895#
3898#	3903#	3907#	3910#	3917#	3925#	3933#	3941#	3946#	3949#	3954#	3960#	3966#	3970#	3973#
3978#	3982#	4234#	4248#	4277#	4390#	4427#	4462#	4546#	4557#	4579#	4645#	4660#	4673#	4688#
4701#	4716#	4722#	4733#	4749#	4755#	4760#	4775#	4780#	4785#	4791#	4799#	4813#	4823#	4842#
4846#	4860#	4863#	4872#	4885#	4891#	4905#	4909#	4915#	4920#	4926#	4943#	4962#	4988#	5014#
5015#	5020#	5023#	5024#	5034#	5039#	5042#	5045#	5047#	5051#	5053#	5054#	5059#	5068#	5075#
5077#	5085#	5092#	5095#	5098#	5101#	5117#	5118#	5123#	5126#	5127#	5137#	5142#	5145#	5148#
5150#	5154#	5156#	5157#	5162#	5171#	5178#	5180#	5188#	5195#	5197#	5205#	5212#	5214#	5222#
5229#	5231#	5239#	5246#	5249#	5252#	5255#	5269#	5270#	5275#	5283#	5288#	5291#	5294#	5296#
5299#	5303#	5317#	5318#	5323#	5326#	5333#	5338#	5341#	5344#	5346#	5350#	5353#	5356#	5376#
5377#	5381#	5382#	5395#	5400#	5405#	5408#	5409#	5414#	5424#	5431#	5442#	5448#	5455#	5468#
5479#	5486#	5497#	5503#	5510#	5518#	5528#	5535#	5537#	5547#	5554#	5557#	5560#	5581#	5582#
5587#	5595#	5602#	5607#	5625#	5626#	5629#	5635#	5642#	5645#	5651#	5658#	5661#	5688#	5689#
5690#	5691#	5704#	5711#	5713#	5714#	5726#	5733#	5735#	5736#	5751#	5756#	5760#	5768#	5775#
5778#	5781#	5801#	5802#	5805#	5810#	5822#	5846#	5851#	5859#	5866#	6221#	6222#	6223#	6224#
6229#	6242#	6247#	6252#	6254#	6262#	6269#	6272#	6274#	6275#	6288#	6308#	6313#	6318#	6320#
6328#	6335#	6348#	6355#	6358#	6397#	6398#	6400#	6401#	6406#	6439#	6444#	6449#	6451#	6459#
6466#	6468#	6482#	6487#	6492#	6494#	6502#	6509#	6512#	6514#	6515#	6520#	6543#	6548#	6553#
6555#	6560#	6567#	6577#	6586#	6597#	6636#	6637#	6641#	6669#	6674#	6679#	6681#	6689#	6694#
6699#	6701#	6716#	6721#	6726#	6728#	6736#	6741#	6746#	6751#	6764#	6774#	6781#	6790#	6820#
6821#	6827#	6843#	6851#	6859#	6867#	6871#	6876#	6892#	6899#	6919#	6926#	6933#	6936#	6964#
6965#	6970#	6976#	6990#	6997#	7002#	7009#	7018#	7023#	7034#	7041#	7044#	7049#	7056#	7093#
7094#	7100#	7109#	7120#	7125#	7130#	7135#	7142#	7146#	7162#	7166#	7171#	7182#	7194#	7206#
7213#	7218#	7233#	7240#	7247#	7275#	7276#	7282#	7291#	7301#	7306#	7311#	7316#	7321#	7329#
7333#	7338#	7345#	7350#	7366#	7373#	7380#	7383#	7421#	7422#	7428#	7437#	7447#	7452#	7457#
7462#	7467#	7476#	7480#	7485#	7501#	7506#	7510#	7515#	7533#	7540#	7552#	7559#	7562#	7586#
7587#	7593#	7602#	7612#	7617#	7622#	7627#	7632#	7646#	7650#	7655#	7665#	7670#	7674#	7679#
7691#	7697#	7704#	7709#	7715#	7745#	7746#	7752#	7761#	7771#	7776#	7781#	7786#	7791#	7800#
7804#	7809#	7818#	7823#	7827#	7832#	7846#	7853#	7860#	7865#	7874#	7898#	7899#	7905#	7921#
7929#	7937#	7946#	7951#	7956#	7965#	7970#	7974#	7991#	8003#	8006#	8033#	8034#	8040#	8049#
8059#	8064#	8069#	8074#	8079#	8086#	8090#	8095#	8104#	8111#	8116#	8120#	8125#	8133#	8140#
8143#	8171#	8172#	8178#	8187#	8197#	8202#	8207#	8212#	8217#	8224#	8228#	8233#	8242#	8249#
8254#	8258#	8263#	8271#	8278#	8281#	8319#	8320#	8326#	8335#	8345#	8350#	8355#	8360#	8365#
8374#	8378#	8383#	8397#	8402#	8406#	8411#	8424#	8434#	8441#	8444#	8481#	8482#	8488#	8497#
8507#	8512#	8517#	8522#	8527#	8535#	8539#	8544#	8558#	8563#	8567#	8572#	8585#	8595#	8602#
8605#	8634#	8635#	8641#	8650#	8660#	8665#	8670#	8675#	8680#	8693#	8701#	8706#	8715#	8720#
8724#	8729#	8742#	8752#	8759#	8764#	8811#	8812#	8818#	8827#	8837#	8842#	8847#	8852#	8857#
8864#	8868#	8873#	8886#	8891#	8895#	8900#	8916#	8929#	8938#	8950#	8959#	8968#	8978#	8985#
8988#	9038#	9039#	9045#	9054#	9064#	9069#	9074#	9079#	9084#	9096#	9104#	9109#	9118#	9123#
9127#	9132#	9150#	9157#	9164#	9169#	9180#	9207#	9208#	9214#	9223#	9233#	9238#	9243#	9248#
9253#	9265#	9275#	9278#	9287#	9292#	9296#	9301#	9320#	9327#	9334#	9339#	9350#	9380#	9381#
9387#	9396#	9406#	9411#	9416#	9421#	9426#	9433#	9437#	9442#	9452#	9457#	9461#	9466#	9474#
9484#	9491#	9494#	9545#	9546#	9552#	9561#	9571#	9576#	9581#	9586#	9591#	9607#	9614#	9619#
9628#	9633#	9637#	9642#	9657#	9664#	9671#	9676#	9689#	9734#	9735#	9741#	9750#	9760#	9765#
9770#	9775#	9780#	9792#	9800#	9805#	9814#	9819#	9823#	9828#	9838#	9845#	9852#	9857#	9872#
9897#	9898#	9904#	9913#	9923#	9928#	9933#	9938#	9943#	9959#	9963#	9968#	9977#	9982#	9986#
9991#	10000#	10007#	10014#	10019#	10033#	10136#	10137#	10143#	10152#	10162#	10167#	10172#	10177#	10182#

```
         10205#  10214#  10219#  10228#  10233#  10237#  10242#  10254#  10270#  10284#  10291#  10294#  10359#  10360#  10366#
         10375#  10385#  10390#  10395#  10400#  10405#  10416#  10424#  10429#  10447#  10452#  10455#  10466#  10473#  10483#
         10514#  10515#  10521#  10530#  10540#  10545#  10550#  10555#  10560#  10566#  10570#  10575#  10586#  10591#  10595#
         10600#  10609#  10619#  10626#  10629#  10674#  10675#  10681#  10690#  10700#  10705#  10710#  10715#  10720#  10732#
         10736#  10741#  10751#  10756#  10760#  10765#  10773#  10780#  10785#  10789#  10794#  10799#  10808#  10841#  10842#
         10848#  10857#  10867#  10872#  10877#  10882#  10887#  10892#  10896#  10901#  10918#  10923#  10927#  10932#  10941#
         10951#  10958#  10961#  11013#  11014#  11020#  11029#  11039#  11044#  11049#  11054#  11059#  11068#  11075#  11080#
         11094#  11099#  11103#  11108#  11116#  11126#  11134#  11142#  11181#  11182#  11188#  11197#  11207#  11212#  11217#
         11222#  11227#  11241#  11245#  11250#  11264#  11269#  11273#  11278#  11289#  11297#  11307#  11314#  11323#  11370#
         11371#  11377#  11386#  11396#  11401#  11406#  11411#  11416#  11428#  11438#  11448#  11453#  11462#  11467#  11471#
         11476#  11485#  11495#  11502#  11515#  11525#  11578#  11579#  11585#  11594#  11604#  11609#  11614#  11619#  11624#
         11634#  11647#  11652#  11661#  11666#  11670#  11675#  11687#  11702#  11713#  11720#  11727#  11732#  11745#  11793#
         11794#  11800#  11809#  11819#  11824#  11829#  11834#  11839#  11845#  11858#  11863#  11872#  11877#  11881#  11886#
         11895#  11910#  11921#  11928#  11935#  11942#  12029#  12030#  12036#  12045#  12055#  12060#  12065#  12070#  12075#
         12091#  12099#  12104#  12113#  12118#  12122#  12127#  12144#  12155#  12162#  12167#  12176#  12181#  12190#  12196#
         12201#  12216#  12227#  12236#  12241#  12248#  12253#  12262#  12267#  12271#  12276#  12295#  12305#  12312#  12315#
         12372#  12373#  12379#  12388#  12398#  12403#  12408#  12413#  12418#  12425#  12432#  12437#  12446#  12451#  12455#
         12460#  12469#  12479#  12486#  12492#  12499#  12504#  12513#  12518#  12522#  12527#  12537#  12542#  12550#  12557#
         12566#  12619#  12620#  12626#  12635#  12645#  12650#  12655#  12660#  12665#  12680#  12684#  12689#  12698#  12703#
         12707#  12712#  12720#  12725#  12751#  12756#  12763#  12766#  12786#  12787#  12790#  12796#  12812#  12817#  12825#
         12830#  12846#  12851#  12859#  12864#  12890#  12895#  12903#  12908#  12935#  12940#  12947#  12952#  12981#  12988#
         12995#  13002#  13009#  13015#  13139#
MSIOSE       1#   1416#
MSLDRO       1#   1416#   4233#   4247#   4754#   4759#   4774#   4779#   4790#   4822#   5041#   5044#   5097#   5144#   5147#
          5251#   5290#   5293#   5298#   5340#   5343#   5352#   7008#   7043#   7181#
MSMASK       1#   1416#
MSMCHI       1#   1416#
MSMCLO       1#   1416#
MSMSK1       1#   1416#
MSPOP        1#   1416#   1590#   2924#   2940#   2961#   2974#   2987#   3004#   3023#   3040#   3053#   3074#   3100#   3113#
          3125#   3138#   3150#   3175#   3188#   3208#   3222#   3237#   3265#   3293#   3307#   3336#   3363#   3390#   3425#
          3437#   3475#   3523#   3542#   3562#   3582#   3610#   3629#   3670#   3697#   3710#   3730#   3757#   3769#   3781#
          3793#   3805#   3832#   3852#   3894#   3906#   3945#   3969#   3981#   4721#   4740#   4845#   4862#   4904#   4914#
          4925#   4946#   4966#   4994#   5050#   5074#   5091#   5094#   5100#   5153#   5177#   5194#   5211#   5228#   5245#
          5248#   5254#   5302#   5349#   5355#   5404#   5430#   5454#   5485#   5509#   5534#   5553#   5556#   5559#   5601#
          5606#   5641#   5657#   5660#   5710#   5732#   5774#   5777#   5780#   5865#   6251#   6268#   6271#   6317#   6334#
          6354#   6357#   6448#   6465#   6491#   6508#   6511#   6552#   6566#   6585#   6596#   6678#   6698#   6725#   6745#
          6780#   6789#   6858#   6932#   6935#   6996#   7040#   7055#   7145#   7212#   7239#   7246#   7320#   7379#   7382#
          7466#   7558#   7561#   7631#   7696#   7714#   7790#   7852#   7873#   7936#   8002#   8005#   8078#   8139#   8142#
          8216#   8277#   8280#   8364#   8440#   8443#   8526#   8601#   8604#   8679#   8758#   8763#   8856#   8984#   8987#
          9083#   9156#   9179#   9252#   9326#   9349#   9425#   9490#   9493#   9590#   9663#   9688#   9779#   9844#   9871#
          9942#  10006#  10032#  10181#  10290#  10293#  10404#  10472#  10482#  10559#  10625#  10628#  10719#  10779#  10798#
         10807#  10886#  10957#  10960#  11058#  11133#  11141#  11226#  11313#  11322#  11415#  11501#  11514#  11524#  11623#
         11719#  11744#  11838#  11927#  11941#  12074#  12161#  12235#  12311#  12314#  12417#  12485#  12556#  12565#  12664#
         12762#  12765#  13014#  13152#  13179#
MSPRIN       1#   1416#   2917#   2930#   2946#   2953#   2967#   2980#   2993#   3010#   3016#   3029#   3046#   3059#   3065#
          3080#   3086#   3093#   3106#   3119#   3131#   3144#   3156#   3162#   3168#   3181#   3194#   3201#   3214#   3228#
          3243#   3250#   3257#   3271#   3278#   3285#   3299#   3313#   3320#   3328#   3342#   3348#   3355#   3369#   3375#
          3382#   3397#   3405#   3415#   3431#   3443#   3449#   3458#   3465#   3483#   3490#   3496#   3502#   3509#   3516#
          3529#   3535#   3548#   3554#   3568#   3574#   3588#   3596#   3603#   3616#   3622#   3635#   3641#   3648#   3655#
          3662#   3676#   3682#   3690#   3703#   3716#   3723#   3736#   3743#   3749#   3763#   3775#   3787#   3799#   3811#
          3817#   3824#   3838#   3844#   3858#   3865#   3872#   3879#   3886#   3900#   3914#   3922#   3930#   3938#   3951#
          3957#   3963#   3975#   4386#   4422#   4539#   4554#   4575#   4641#   4656#   4669#   4684#   4697#   4796#   4882#
         12977#  12984#  12991#  12998#  13005#
MSPUSH       1#   1416#   1420#   1582#   2915#   2928#   2944#   2965#   2978#   2991#   3008#   3027#   3044#   3057#   3078#
          3104#   3117#   3129#   3142#   3154#   3179#   3192#   3212#   3226#   3241#   3269#   3297#   3311#   3340#   3367#
```

```
          3394#   3429#   3441#   3479#   3527#   3546#   3566#   3586#   3614#   3633#   3674#   3701#   3714#   3734#   3761#
          3773#   3785#   3797#   3809#   3836#   3856#   3898#   3910#   3949#   3973#   4716#   4733#   4749#   4860#   4872#
          4909#   4920#   4943#   4962#   4988#   5014#   5015    5023#   5024    5053#   5054    5059#   5077#   5117#   5118
          5126#   5127    5156#   5157    5162#   5180#   5197#   5214#   5231#   5269#   5270    5317#   5318    5326#   5376#
          5377    5381#   5382    5408#   5409    5414#   5442#   5468#   5497#   5518#   5537#   5581#   5582    5587#   5625#
          5626    5629#   5645#   5688#   5689    5690#   5691    5713#   5714    5735#   5736    5760#   5801#   5802    6221#
          6222    6223#   6224    6229#   6254#   6274#   6275    6288#   6320#   6397#   6398    6400#   6401    6406#   6451#
          6468#   6494#   6514#   6515    6520#   6555#   6636#   6637    6641#   6681#   6701#   6728#   6751#   6820#   6821
          6827#   6867#   6964#   6965    6970#   7002#   7093#   7094    7100#   7162#   7218#   7275#   7276    7282#   7329#
          7421#   7422    7428#   7476#   7586#   7587    7593#   7646#   7745#   7746    7752#   7800#   7898#   7899    7905#
          7946#   8033#   8034    8040#   8086#   8171#   8172    8178#   8224#   8319#   8320    8326#   8374#   8481#   8482
          8488#   8535#   8634#   8635    8641#   8693#   8811#   8812    8818#   8864#   9038#   9039    9045#   9096#   9207#
          9208    9214#   9265#   9380#   9381    9387#   9433#   9545#   9546    9552#   9607#   9734#   9735    9741#   9792#
          9897#   9898    9904#   9959#   10136#  10137   10143#  10205#  10359#  10360   10366#  10416#  10514#  10515   10521#
          10566#  10674#  10675   10681#  10732#  10785#  10841#  10842   10848#  10892#  11013#  11014   11020#  11068#  11181#
          11182   11188#  11241#  11370#  11371   11377#  11428#  11438#  11578#  11579   11585#  11634#  11793#  11794   11800#
          11845#  12029#  12030   12036#  12091#  12167#  12241#  12372#  12373   12379#  12425#  12492#  12619#  12620   12626#
          12680#  12786#  12787   13139#
MSPUT     1#      1416#   2917#   2930#   2946#   2953#   2967#   2980#   2993#   3010#   3016#   3029#   3046#   3059#   3065#
          3080#   3086#   3093#   3106#   3119#   3131#   3144#   3156#   3162#   3168#   3181#   3194#   3201#   3214#   3228#
          3243#   3250#   3257#   3271#   3278#   3285#   3299#   3313#   3320#   3328#   3342#   3348#   3355#   3369#   3375#
          3382#   3397#   3405#   3415#   3431#   3443#   3449#   3458#   3465#   3483#   3490#   3496#   3502#   3509#   3516#
          3529#   3535#   3548#   3554#   3568#   3574#   3588#   3596#   3603#   3616#   3622#   3635#   3641#   3648#   3655#
          3662#   3676#   3682#   3690#   3703#   3716#   3723#   3736#   3743#   3749#   3763#   3775#   3787#   3799#   3811#
          3817#   3824#   3838#   3844#   3858#   3865#   3872#   3879#   3886#   3900#   3914#   3922#   3930#   3938#   3951#
          3957#   3963#   3975#   4386#   4422#   4539#   4554#   4575#   4641#   4656#   4669#   4684#   4697#   4796#   4809#
          4882#   5016#   5119#   5271#   5319#   6972#   7138#   12977#  12984#  12991#  12998#  13005#
MSPUT1    1#      1416#   2917#   2918    2919    2930#   2931    2932    2933    2934    2946#   2947    2948    2953#   2954
          2955    2956    2967#   2968    2980#   2981    2993#   2994    2995    2996    2997    2998    3010#   3011    3016#
          3017    3018    3029#   3030    3031    3032    3033    3034    3046#   3047    3059#   3060    3065#   3066    3067
          3068    3069    3080#   3081    3086#   3087    3088    3093#   3094    3095    3106#   3107    3119#   3120    3131#
          3132    3133    3144#   3145    3156#   3157    3162#   3163    3168#   3169    3170    3181#   3182    3183    3194#
          3195    3196    3201#   3202    3203    3214#   3215    3216    3217    3228#   3229    3230    3231    3232    3243#
          3244    3245    3250#   3251    3252    3257#   3258    3259    3260    3271#   3272    3273    3278#   3279    3280
          3285#   3286    3287    3288    3299#   3300    3301    3302    3313#   3314    3315    3320#   3321    3322    3323
          3328#   3329    3330    3331    3342#   3343    3348#   3349    3350    3355#   3356    3357    3358    3369#   3370
          3375#   3376    3377    3382#   3383    3384    3385    3397#   3398    3399    3405#   3406    3407    3408    3409
          3415#   3416    3417    3418    3419    3431#   3432    3443#   3444    3449#   3450    3458#   3459    3460    3465#
          3466    3467    3483#   3484    3490#   3491    3496#   3497    3502#   3503    3504    3509#   3510    3511    3516#
          3517    3518    3529#   3530    3535#   3536    3537    3548#   3549    3554#   3555    3556    3557    3568#   3569
          3574#   3575    3576    3577    3588#   3589    3590    3591    3596#   3597    3598    3603#   3604    3605    3616#
          3617    3622#   3623    3624    3635#   3636    3641#   3642    3643    3648#   3649    3650    3655#   3656    3657
          3662#   3663    3664    3665    3676#   3677    3682#   3683    3684    3685    3690#   3691    3692    3703#   3704
          3705    3716#   3717    3718    3723#   3724    3725    3736#   3737    3738    3743#   3744    3749#   3750    3751
          3752    3763#   3764    3775#   3776    3787#   3788    3799#   3800    3811#   3812    3817#   3818    3819    3824#
          3825    3826    3827    3838#   3839    3844#   3845    3846    3847    3858#   3859    3860    3865#   3866    3867
          3872#   3873    3874    3879#   3880    3881    3886#   3887    3888    3889    3900#   3901    3914#   3915    3922#
          3923    3930#   3931    3938#   3939    3951#   3952    3957#   3958    3963#   3964    3975#   3976    4386#   4387
          4388    4422#   4423    4424    4425    4539#   4540    4541    4542    4543    4544    4554#   4555    4575#   4576
          4577    4641#   4642    4643    4656#   4657    4658    4669#   4670    4671    4684#   4685    4686    4697#   4698
          4699    4796#   4797    4809#   4810    4811    4812    4882#   4883    5016#   5017    5018    5019    5119#   5120
          5121    5122    5271#   5272    5273    5274    5319#   5320    5321    5322    6972#   6973    6974    6975    7138#
          7139    7140    7141    12977#  12978   12979   12984#  12985   12986   12991#  12992   12993   12998#  12999   13000
          13005#  13006   13007
MSRADI    1#      1416#   13142#  13147#
MSRBRO    1#      1416#
```

```
MSRNRO        1#     1416#    4785#    4786     4790#    4792     4822#    4824
MSSETS        1#     1416#    1420#    1582#    2915#    2928#    2944#    2965#    2978#    2991#    3008#    3027#    3044#    3057#    3078#
           3104#     3117#    3129#    3142#    3154#    3179#    3192#    3212#    3226#    3241#    3269#    3297#    3311#    3340#    3367#
           3394#     3429#    3441#    3479#    3527#    3546#    3566#    3586#    3614#    3633#    3674#    3701#    3714#    3734#    3761#
           3773#     3785#    3797#    3809#    3836#    3856#    3898#    3910#    3949#    3973#    4716#    4733#    4749#    4860#    4872#
           4909#     4920#    4943#    4962#    4988#    5015#    5024#    5054#    5059#    5077#    5118#    5127#    5157#    5162#    5180#
           5197#     5214#    5231#    5270#    5318#    5326#    5377#    5382#    5409#    5414#    5442#    5468#    5497#    5518#    5537#
           5582#     5587#    5626#    5629#    5645#    5689#    5691#    5714#    5736#    5760#    5802#    6222#    6224#    6229#    6254#
           6275#     6288#    6320#    6398#    6401#    6406#    6451#    6468#    6494#    6515#    6520#    6555#    6637#    6641#    6681#
           6701#     6728#    6751#    6821#    6827#    6867#    6965#    6970#    7002#    7094#    7100#    7162#    7218#    7276#    7282#
           7329#     7422#    7428#    7476#    7587#    7593#    7646#    7746#    7752#    7800#    7899#    7905#    7946#    8034#    8040#
           8086#     8172#    8178#    8224#    8320#    8326#    8374#    8482#    8488#    8535#    8635#    8641#    8693#    8812#    8818#
           8864#     9039#    9045#    9096#    9208#    9214#    9265#    9381#    9387#    9433#    9546#    9552#    9607#    9735#    9741#
           9792#     9898#    9904#    9959#   10137#   10143#   10205#   10360#   10366#   10416#   10515#   10521#   10566#   10675#   10681#
          10732#    10785#   10842#   10848#   10892#   11014#   11020#   11068#   11182#   11188#   11241#   11371#   11377#   11428#   11438#
          11579#    11585#   11634#   11794#   11800#   11845#   12030#   12036#   12091#   12167#   12241#   12373#   12379#   12425#   12492#
          12620#    12626#   12680#   12787#   13139#
MSSTAR        1#     1416#
MSSVC         1#     1416#    2917#    2921     2924#    2925     2930#    2936     2940#    2941     2946#    2950     2953#    2958     2961#
           2962      2967#    2970     2974#    2975     2980#    2983     2987#    2988     2993#    3000     3004#    3005     3010#    3013
           3016#     3020     3023#    3024     3029#    3036     3040#    3041     3046#    3049     3053#    3054     3059#    3062     3065#
           3071      3074#    3075     3080#    3083     3086#    3090     3093#    3097     3100#    3101     3106#    3109     3113#    3114
           3119#     3122     3125#    3126     3131#    3135     3138#    3139     3144#    3147     3150#    3151     3156#    3159     3162#
           3165      3168#    3172     3175#    3176     3181#    3185     3188#    3189     3194#    3198     3201#    3205     3208#    3209
           3214#     3219     3222#    3223     3228#    3234     3237#    3238     3243#    3247     3250#    3254     3257#    3262     3265#
           3266      3271#    3275     3278#    3282     3285#    3290     3293#    3294     3299#    3304     3307#    3308     3313#    3317
           3320#     3325     3328#    3333     3336#    3337     3342#    3345     3348#    3352     3355#    3360     3363#    3364     3369#
           3372      3375#    3379     3382#    3387     3390#    3391     3397#    3401     3405#    3411     3415#    3421     3425#    3426
           3431#     3434     3437#    3438     3443#    3446     3449#    3452     3458#    3462     3465#    3469     3475#    3476     3483#
           3486      3490#    3493     3496#    3499     3502#    3506     3509#    3513     3516#    3520     3523#    3524     3529#    3532
           3535#     3539     3542#    3543     3548#    3551     3554#    3559     3562#    3563     3568#    3571     3574#    3579     3582#
           3583      3588#    3593     3596#    3600     3603#    3607     3610#    3611     3616#    3619     3622#    3626     3629#    3630
           3635#     3638     3641#    3645     3648#    3652     3655#    3659     3662#    3667     3670#    3671     3676#    3679     3682#
           3687      3690#    3694     3697#    3698     3703#    3707     3710#    3711     3716#    3720     3723#    3727     3730#    3731
           3736#     3740     3743#    3746     3749#    3754     3757#    3758     3763#    3766     3769#    3770     3775#    3778     3781#
           3782      3787#    3790     3793#    3794     3799#    3802     3805#    3806     3811#    3814     3817#    3821     3824#    3829
           3832#     3833     3838#    3841     3844#    3849     3852#    3853     3858#    3862     3865#    3869     3872#    3876     3879#
           3883      3886#    3891     3894#    3895     3900#    3903     3906#    3907     3914#    3917     3922#    3925     3930#    3933
           3938#     3941     3945#    3946     3951#    3954     3957#    3960     3963#    3966     3969#    3970     3975#    3978     3981#
           3982      4233#    4234     4247#    4248     4277#    4386#    4390     4422#    4427     4462#    4539#    4546     4554#    4557
           4575#     4579     4641#    4645     4656#    4660     4669#    4673     4684#    4688     4697#    4701     4718#    4721#    4722
           4754#     4755     4759#    4760     4774#    4775     4779#    4780     4785#    4790#    4791     4796#    4799     4809#    4813
           4822#     4823     4842#    4845#    4846     4862#    4863     4882#    4885     4891#    4904#    4905     4911#    4914#    4915
           4922#     4925#    4926     5016#    5020     5023#    5024     5034     5039#    5041#    5042     5044#    5045     5047#    5050#
           5051      5053#    5054     5059#    5068     5074#    5075     5077#    5085     5091#    5092     5094#    5095     5097#    5098
           5100#     5101     5119#    5123     5126#    5127     5137     5142#    5144#    5145     5147#    5148     5150#    5153#    5154
           5156#     5157     5162#    5171     5177#    5178     5180#    5188     5194#    5195     5197#    5205     5211#    5212     5214#
           5222      5228#    5229     5231#    5239     5245#    5246     5248#    5249     5251#    5252     5254#    5255     5271#    5275
           5283      5288#    5290#    5291     5293#    5294     5296#    5298#    5299     5302#    5303     5319#    5323     5326#    5333
           5338#     5340#    5341     5343#    5344     5346#    5349#    5350     5352#    5353     5355#    5356     5381#    5382     5395
           5400#     5404#    5405     5408#    5409     5414#    5424     5430#    5431     5442#    5448     5454#    5455     5468#    5479
           5485#     5486     5497#    5503     5509#    5510     5518#    5528     5534#    5535     5537#    5547     5553#    5554     5556#
           5557      5559#    5560     5587#    5595     5601#    5602     5606#    5607     5629#    5635     5641#    5642     5645#    5651
           5657#     5658     5660#    5661     5690#    5691     5704     5710#    5711     5713#    5714     5726     5732#    5733     5735#
           5736      5751     5756#    5760#    5768     5774#    5775     5777#    5778     5780#    5781     5805     5810#    5822     5846
           5851#     5859     5865#    5866     6223#    6224     6229#    6242     6247#    6251#    6252     6254#    6262     6268#    6269
```

6271#	6272	6274#	6275	6288#	6308	6313#	6317#	6318	6320#	6328	6334#	6335	6348	6354#
6355	6357#	6358	6400#	6401	6406#	6439	6444#	6448#	6449	6451#	6459	6465#	6466	6468#
6482	6487#	6491#	6492	6494#	6502	6508#	6509	6511#	6512	6514#	6515	6520#	6543	6548#
6552#	6553	6555#	6560	6566#	6567	6577	6585#	6586	6596#	6597	6641#	6669	6674#	6678#
6679	6681#	6689	6694#	6698#	6699	6701#	6716	6721#	6725#	6726	6728#	6736	6741#	6745#
6746	6751#	6764	6774	6780#	6781	6789#	6790	6827#	6843	6851	6858#	6859	6867#	6871
6876#	6892	6899	6919	6926	6932#	6933	6935#	6936	6970#	6972#	6976	6990	6996#	6997
7002#	7008#	7009	7018	7023#	7034	7040#	7041	7043#	7044	7049	7055#	7056	7100#	7109#
7120	7125#	7130	7135#	7138#	7142	7145#	7146	7162#	7166	7171#	7181#	7182	7194	7206
7212#	7213	7218#	7233	7239#	7240	7246#	7247	7282#	7291#	7301	7306#	7311	7316#	7320#
7321	7329#	7333	7338#	7345	7350#	7366	7373	7379#	7380	7382#	7383	7428#	7437#	7447
7452#	7457	7462#	7466#	7467	7476#	7480	7485#	7501	7506#	7510	7515#	7533	7540	7552
7558#	7559	7561#	7562	7593#	7602#	7612	7617#	7622	7627#	7631#	7632	7646#	7650	7655#
7665	7670#	7674	7679#	7691	7696#	7697	7704	7709#	7714#	7715	7752#	7761#	7771	7776#
7781	7786#	7790#	7791	7800#	7804	7809#	7818	7823#	7827	7832#	7846	7852#	7853	7860
7865#	7873#	7874	7905#	7921	7929	7936#	7937	7946#	7951	7956#	7965	7970#	7974	7991
8002#	8003	8005#	8006	8040#	8049#	8059	8064#	8069	8074#	8078#	8079	8086#	8090	8095#
8104	8111	8116#	8120	8125#	8133	8139#	8140	8142#	8143	8178#	8187#	8197	8202#	8207
8212#	8216#	8217	8224#	8228	8233#	8242	8249	8254#	8258	8263#	8271	8277#	8278	8280#
8281	8326#	8335#	8345	8350#	8355	8360#	8364#	8365	8374#	8378	8383#	8397	8402#	8406
8411#	8424	8434	8440#	8441	8443#	8444	8488#	8497#	8507	8512#	8517	8522#	8526#	8527
8535#	8539	8544#	8558	8563#	8567	8572#	8585	8595	8601#	8602	8604#	8605	8641#	8650#
8660	8665#	8670	8675#	8679#	8680	8693#	8701	8706#	8715	8720#	8724	8729#	8742	8752
8758#	8759	8763#	8764	8818#	8827#	8837	8842#	8847	8852#	8856#	8857	8864#	8868	8873#
8886	8891#	8895	8900#	8916	8929	8938	8950	8959	8968	8978	8984#	8985	8987#	8988
9045#	9054#	9064	9069#	9074	9079#	9083#	9084	9096#	9104	9109#	9118	9123#	9127	9132#
9150	9156#	9157	9164	9169#	9179#	9180	9214#	9223#	9233	9238#	9243	9248#	9252#	9253
9265#	9273	9278#	9287	9292#	9296	9301#	9320	9326#	9327	9334	9339#	9349#	9350	9387#
9396#	9406	9411#	9416	9421#	9425#	9426	9433#	9437	9442#	9452	9457#	9461	9466#	9474
9484	9490#	9491	9493#	9494	9552#	9561#	9571	9576#	9581	9586#	9590#	9591	9607#	9614
9619#	9628	9633#	9637	9642#	9657	9663#	9664	9671	9676#	9688#	9689	9741#	9750#	9760
9765#	9770	9775#	9779#	9780	9792#	9800	9805#	9814	9819#	9823	9828#	9838	9844#	9845
9852	9857#	9871#	9872	9904#	9913#	9923	9928#	9933	9938#	9942#	9943	9959#	9963	9968#
9977	9982#	9986	9991#	10000	10006#	10007	10014	10019#	10032#	10033	10143#	10152#	10162	10167#
10172	10177#	10181#	10182	10205#	10214	10219#	10228	10233#	10237	10242#	10254	10270	10284	10290#
10291	10293#	10294	10366#	10375#	10385	10390#	10395	10400#	10404#	10405	10416#	10424	10429#	10447
10452#	10455	10466	10472#	10473	10482#	10483	10521#	10530#	10540	10545#	10550	10555#	10559#	10560
10566#	10570	10575#	10586	10591#	10595	10600#	10609	10619	10625#	10626	10628#	10629	10681#	10690#
10700	10705#	10710	10715#	10719#	10720	10732#	10736	10741#	10751	10756#	10760	10765#	10773	10779#
10780	10785#	10789	10794#	10798#	10799	10807#	10808	10848#	10857#	10867	10872#	10877	10882#	10886#
10887	10892#	10896	10901#	10918	10923#	10927	10932#	10941	10951	10957#	10958	10960#	10961	11020#
11029#	11039	11044#	11049	11054#	11058#	11059	11068#	11075	11080#	11094	11099#	11103	11108#	11116
11126	11133#	11134	11141#	11142	11188#	11197#	11207	11212#	11217	11222#	11226#	11227	11241#	11245
11250#	11264	11269#	11273	11278#	11289	11297	11307	11313#	11314	11322#	11323	11377#	11386#	11396
11401#	11406	11411#	11415#	11416	11428#	11438#	11448	11453#	11462	11467#	11471	11476#	11485	11495
11501#	11502	11514#	11515	11524#	11525	11585#	11594#	11604	11609#	11614	11619#	11623#	11624	11634#
11647	11652#	11661	11666#	11670	11675#	11687	11702	11713	11719#	11720	11727	11732#	11744#	11745
11800#	11809#	11819	11824#	11829	11834#	11838#	11839	11845#	11858	11863#	11872	11877#	11881	11886#
11895	11910	11921	11927#	11928	11935	11941#	11942	12036#	12045#	12055	12060#	12065	12070#	12074#
12075	12091#	12099	12104#	12113	12118#	12122	12127#	12144	12155	12161#	12162	12167#	12176	12181#
12190	12196	12201#	12216	12227	12235#	12236	12241#	12248	12253#	12262	12267#	12271	12276#	12295
12305	12311#	12312	12314#	12315	12379#	12388#	12398	12403#	12408	12413#	12417#	12418	12425#	12432
12437#	12446	12451#	12455	12460#	12469	12479	12485#	12486	12492#	12499	12504#	12513	12518#	12522
12527#	12537	12542#	12550	12556#	12557	12565#	12566	12626#	12635#	12645	12650#	12655	12660#	12664#
12665	12680#	12684	12689#	12698	12703#	12707	12712#	12720	12725#	12751	12756#	12762#	12763	12765#
12766	12790#	12796#	12812	12817#	12825	12830#	12846	12851#	12859	12864#	12890	12895#	12903	12908#

```
        12935  12940#  12947  12952#  12977#  12981  12984#  12988  12991#  12995  12998#  13002  13005#  13009  13014#
        13015
MSTLAB      1#  1416#  2921#  2925#  2936#  2941#  2950#  2958#  2962#  2970#  2975#  2983#  2988#  3000#  3005#
         3013#  3020#  3024#  3036#  3041#  3049#  3054#  3062#  3071#  3075#  3083#  3090#  3097#  3101#  3109#
         3114#  3122#  3126#  3135#  3139#  3147#  3151#  3159#  3165#  3172#  3176#  3185#  3189#  3198#  3205#
         3209#  3219#  3223#  3234#  3238#  3247#  3254#  3262#  3266#  3275#  3282#  3290#  3294#  3304#  3308#
         3317#  3325#  3333#  3337#  3345#  3352#  3360#  3364#  3372#  3379#  3387#  3391#  3401#  3411#  3421#
         3426#  3434#  3438#  3446#  3452#  3462#  3469#  3476#  3486#  3493#  3499#  3506#  3513#  3520#  3524#
         3532#  3539#  3543#  3551#  3559#  3563#  3571#  3579#  3583#  3593#  3600#  3607#  3611#  3619#  3626#
         3630#  3638#  3645#  3652#  3659#  3667#  3671#  3679#  3687#  3694#  3698#  3707#  3711#  3720#  3727#
         3731#  3740#  3746#  3754#  3758#  3766#  3770#  3778#  3782#  3790#  3794#  3802#  3806#  3814#  3821#
         3829#  3833#  3841#  3849#  3853#  3862#  3869#  3876#  3883#  3891#  3895#  3903#  3907#  3917#  3925#
         3933#  3941#  3946#  3954#  3960#  3966#  3970#  3978#  3982#  4234#  4248#  4277#  4390#  4427#  4462#
         4546#  4557#  4579#  4645#  4660#  4673#  4688#  4701#  4722#  4755#  4760#  4775#  4780#  4785#  4791#
         4799#  4813#  4823#  4842#  4846#  4863#  4885#  4891#  4905#  4915#  4926#  5020#  5024#  5034#  5039#
         5042#  5045#  5047#  5051#  5054#  5059#  5068#  5075#  5077#  5085#  5092#  5095#  5098#  5101#  5123#
         5127#  5137#  5142#  5145#  5148#  5150#  5154#  5157#  5162#  5171#  5178#  5180#  5188#  5195#  5197#
         5205#  5212#  5214#  5222#  5229#  5231#  5239#  5246#  5249#  5252#  5255#  5275#  5283#  5288#  5291#
         5294#  5296#  5299#  5303#  5323#  5326#  5333#  5338#  5341#  5344#  5346#  5350#  5353#  5356#  5382#
         5395#  5400#  5405#  5409#  5414#  5424#  5431#  5442#  5448#  5455#  5468#  5479#  5486#  5497#  5503#
         5510#  5518#  5528#  5535#  5537#  5547#  5554#  5557#  5560#  5587#  5595#  5602#  5607#  5629#  5635#
         5642#  5645#  5651#  5658#  5661#  5691#  5704#  5711#  5714#  5726#  5733#  5736#  5751#  5756#  5760#
         5768#  5775#  5778#  5781#  5805#  5810#  5822#  5846#  5851#  5859#  5866#  6224#  6229#  6242#  6247#
         6252#  6254#  6262#  6269#  6272#  6275#  6288#  6308#  6313#  6318#  6320#  6328#  6335#  6348#  6355#
         6358#  6401#  6406#  6439#  6444#  6449#  6451#  6459#  6466#  6468#  6482#  6487#  6492#  6494#  6502#
         6509#  6512#  6515#  6520#  6543#  6548#  6553#  6555#  6560#  6567#  6577#  6586#  6597#  6641#  6669#
         6674#  6679#  6681#  6689#  6694#  6699#  6701#  6716#  6721#  6726#  6728#  6736#  6741#  6746#  6751#
         6764#  6774#  6781#  6790#  6827#  6843#  6851#  6859#  6867#  6871#  6876#  6892#  6899#  6919#  6926#
         6933#  6936#  6970#  6976#  6990#  6997#  7002#  7009#  7018#  7023#  7034#  7041#  7044#  7049#  7056#
         7100#  7109#  7120#  7125#  7130#  7135#  7142#  7146#  7162#  7166#  7171#  7182#  7194#  7206#  7213#
         7218#  7233#  7240#  7247#  7282#  7291#  7301#  7306#  7311#  7316#  7321#  7329#  7333#  7338#  7345#
         7350#  7366#  7373#  7380#  7383#  7428#  7437#  7447#  7452#  7457#  7462#  7467#  7476#  7480#  7485#
         7501#  7506#  7510#  7515#  7533#  7540#  7552#  7559#  7562#  7593#  7602#  7612#  7617#  7622#  7627#
         7632#  7646#  7650#  7655#  7665#  7670#  7674#  7679#  7691#  7697#  7704#  7709#  7715#  7752#  7761#
         7771#  7776#  7781#  7786#  7791#  7800#  7804#  7809#  7818#  7823#  7827#  7832#  7846#  7853#  7860#
         7865#  7874#  7905#  7921#  7929#  7937#  7946#  7951#  7956#  7965#  7970#  7974#  7991#  8003#  8006#
         8040#  8049#  8059#  8064#  8069#  8074#  8079#  8086#  8090#  8095#  8104#  8111#  8116#  8120#  8125#
         8133#  8140#  8143#  8178#  8187#  8197#  8202#  8207#  8212#  8217#  8224#  8228#  8233#  8242#  8249#
         8254#  8258#  8263#  8271#  8278#  8281#  8326#  8335#  8345#  8350#  8355#  8360#  8365#  8374#  8378#
         8383#  8397#  8402#  8406#  8411#  8424#  8434#  8441#  8444#  8488#  8497#  8507#  8512#  8517#  8522#
         8527#  8535#  8539#  8544#  8558#  8563#  8567#  8572#  8585#  8595#  8602#  8605#  8641#  8650#  8660#
         8665#  8670#  8675#  8680#  8693#  8701#  8706#  8715#  8720#  8724#  8729#  8742#  8752#  8759#  8764#
         8818#  8827#  8837#  8842#  8847#  8852#  8857#  8864#  8868#  8873#  8886#  8891#  8895#  8900#  8916#
         8929#  8938#  8950#  8959#  8968#  8978#  8985#  8988#  9045#  9054#  9064#  9069#  9074#  9079#  9084#
         9096#  9104#  9109#  9118#  9123#  9127#  9132#  9150#  9157#  9164#  9169#  9180#  9214#  9223#  9233#
         9238#  9243#  9248#  9253#  9265#  9273#  9278#  9287#  9292#  9296#  9301#  9320#  9327#  9334#  9339#
         9350#  9387#  9396#  9406#  9411#  9416#  9421#  9426#  9433#  9437#  9442#  9452#  9457#  9461#  9466#
         9474#  9484#  9491#  9494#  9552#  9561#  9571#  9576#  9581#  9586#  9591#  9607#  9614#  9619#  9628#
         9633#  9637#  9642#  9657#  9664#  9671#  9676#  9689#  9741#  9750#  9760#  9765#  9770#  9775#  9780#
         9792#  9800#  9805#  9814#  9819#  9823#  9828#  9838#  9845#  9852#  9857#  9872#  9904#  9913#  9923#
         9928#  9933#  9938#  9943#  9959#  9963#  9968#  9977#  9982#  9986#  9991# 10000# 10007# 10014# 10019#
        10033# 10143# 10152# 10162# 10167# 10172# 10177# 10182# 10205# 10214# 10219# 10228# 10233# 10237# 10242#
        10254# 10270# 10284# 10291# 10294# 10366# 10375# 10385# 10390# 10395# 10400# 10405# 10416# 10424# 10429#
        10447# 10452# 10455# 10466# 10473# 10483# 10521# 10530# 10540# 10545# 10550# 10555# 10560# 10566# 10570#
        10575# 10586# 10591# 10595# 10600# 10609# 10619# 10626# 10629# 10681# 10690# 10700# 10705# 10710# 10715#
        10720# 10732# 10736# 10741# 10751# 10756# 10760# 10765# 10773# 10780# 10785# 10789# 10794# 10799# 10808#
```

```
         10848#  10857#  10867#  10872#  10877#  10882#  10887#  10892#  10896#  10901#  10918#  10923#  10927#  10932#  10941#
         10951#  10958#  10961#  11020#  11029#  11039#  11044#  11049#  11054#  11059#  11068#  11075#  11080#  11094#  11099#
         11103#  11108#  11116#  11126#  11134#  11142#  11188#  11197#  11207#  11212#  11217#  11222#  11227#  11241#  11245#
         11250#  11264#  11269#  11273#  11278#  11289#  11297#  11307#  11314#  11323#  11377#  11386#  11396#  11401#  11406#
         11411#  11416#  11428#  11438#  11448#  11453#  11462#  11467#  11471#  11476#  11485#  11495#  11502#  11515#  11525#
         11585#  11594#  11604#  11609#  11614#  11619#  11624#  11634#  11647#  11652#  11661#  11666#  11670#  11675#  11687#
         11702#  11713#  11720#  11727#  11732#  11745#  11800#  11809#  11819#  11824#  11829#  11834#  11839#  11845#  11858#
         11863#  11872#  11877#  11881#  11886#  11895#  11910#  11921#  11928#  11935#  11942#  12036#  12045#  12055#  12060#
         12065#  12070#  12075#  12091#  12099#  12104#  12113#  12118#  12122#  12127#  12144#  12155#  12162#  12167#  12176#
         12181#  12190#  12196#  12201#  12216#  12227#  12236#  12241#  12248#  12253#  12262#  12267#  12271#  12276#  12295#
         12305#  12312#  12315#  12379#  12388#  12398#  12403#  12408#  12413#  12418#  12425#  12432#  12437#  12446#  12451#
         12455#  12460#  12469#  12479#  12486#  12492#  12499#  12504#  12513#  12518#  12522#  12527#  12537#  12542#  12550#
         12557#  12566#  12626#  12635#  12645#  12650#  12655#  12660#  12665#  12680#  12684#  12689#  12698#  12703#  12707#
         12712#  12720#  12725#  12751#  12756#  12763#  12766#  12790#  12796#  12812#  12817#  12825#  12830#  12846#  12851#
         12859#  12864#  12890#  12895#  12903#  12908#  12935#  12940#  12947#  12952#  12981#  12988#  12995#  13002#  13009#
         13015#
MSTSTL       1#   1416#   2921#   2925#   2936#   2941#   2950#   2958#   2962#   2970#   2975#   2983#   2988#   3000#   3005#
          3013#   3020#   3024#   3036#   3041#   3049#   3054#   3062#   3071#   3075#   3083#   3090#   3097#   3101#   3109#
          3114#   3122#   3126#   3135#   3139#   3147#   3151#   3159#   3165#   3172#   3176#   3185#   3189#   3198#   3205#
          3209#   3219#   3223#   3234#   3238#   3247#   3254#   3262#   3266#   3275#   3282#   3290#   3294#   3304#   3308#
          3317#   3325#   3333#   3337#   3345#   3352#   3360#   3364#   3372#   3379#   3387#   3391#   3401#   3411#   3421#
          3426#   3434#   3438#   3446#   3452#   3462#   3469#   3476#   3486#   3493#   3499#   3506#   3513#   3520#   3524#
          3532#   3539#   3543#   3551#   3559#   3563#   3571#   3579#   3583#   3593#   3600#   3607#   3611#   3619#   3626#
          3630#   3638#   3645#   3652#   3659#   3667#   3671#   3679#   3687#   3694#   3698#   3707#   3711#   3720#   3727#
          3731#   3740#   3746#   3754#   3758#   3766#   3770#   3778#   3782#   3790#   3794#   3802#   3806#   3814#   3821#
          3829#   3833#   3841#   3849#   3853#   3862#   3869#   3876#   3883#   3891#   3895#   3903#   3907#   3917#   3925#
          3933#   3941#   3946#   3954#   3960#   3966#   3970#   3978#   3982#   4234#   4248#   4277#   4390#   4427#   4462#
          4546#   4557#   4579#   4645#   4660#   4673#   4688#   4701#   4722#   4755#   4760#   4775#   4780#   4785#   4791#
          4799#   4813#   4823#   4842#   4846#   4863#   4885#   4891#   4905#   4915#   4926#   5020#   5024#   5034#   5039#
          5042#   5045#   5047#   5051#   5054#   5059#   5068#   5075#   5077#   5085#   5092#   5095#   5098#   5101#   5123#
          5127#   5137#   5142#   5145#   5148#   5150#   5154#   5157#   5162#   5171#   5178#   5180#   5188#   5195#   5197#
          5205#   5212#   5214#   5222#   5229#   5231#   5239#   5246#   5249#   5252#   5255#   5275#   5283#   5288#   5291#
          5294#   5296#   5299#   5303#   5323#   5326#   5333#   5338#   5341#   5344#   5346#   5350#   5353#   5356#   5382#
          5395#   5400#   5405#   5409#   5414#   5424#   5431#   5442#   5448#   5455#   5468#   5479#   5486#   5497#   5503#
          5510#   5518#   5528#   5535#   5537#   5547#   5554#   5557#   5560#   5587#   5595#   5602#   5607#   5629#   5635#
          5642#   5645#   5651#   5658#   5661#   5691#   5704#   5711#   5714#   5726#   5733#   5736#   5751#   5756#   5760#
          5768#   5775#   5778#   5781#   5805#   5810#   5822#   5846#   5851#   5859#   5866#   6224#   6229#   6242#   6247#
          6252#   6254#   6262#   6269#   6272#   6275#   6288#   6308#   6313#   6318#   6320#   6328#   6335#   6348#   6355#
          6358#   6401#   6406#   6439#   6444#   6449#   6451#   6459#   6466#   6468#   6482#   6487#   6492#   6494#   6502#
          6509#   6512#   6515#   6520#   6543#   6548#   6553#   6555#   6560#   6567#   6577#   6586#   6597#   6641#   6669#
          6674#   6679#   6681#   6689#   6694#   6699#   6701#   6716#   6721#   6726#   6728#   6736#   6741#   6746#   6751#
          6764#   6774#   6781#   6790#   6827#   6843#   6851#   6859#   6867#   6871#   6876#   6892#   6899#   6919#   6926#
          6933#   6936#   6970#   6976#   6990#   6997#   7002#   7009#   7018#   7023#   7034#   7041#   7044#   7049#   7056#
          7100#   7109#   7120#   7125#   7130#   7135#   7142#   7146#   7162#   7166#   7171#   7182#   7194#   7206#   7213#
          7218#   7233#   7240#   7247#   7282#   7291#   7301#   7306#   7311#   7316#   7321#   7329#   7333#   7338#   7345#
          7350#   7366#   7373#   7380#   7383#   7428#   7437#   7447#   7452#   7457#   7462#   7467#   7476#   7480#   7485#
          7501#   7506#   7510#   7515#   7533#   7540#   7552#   7559#   7562#   7593#   7602#   7612#   7617#   7622#   7627#
          7632#   7646#   7650#   7655#   7665#   7670#   7674#   7679#   7691#   7697#   7704#   7709#   7715#   7752#   7761#
          7771#   7776#   7781#   7786#   7791#   7800#   7804#   7809#   7818#   7823#   7827#   7832#   7846#   7853#   7860#
          7865#   7874#   7905#   7921#   7929#   7937#   7946#   7951#   7956#   7965#   7970#   7974#   7991#   8003#   8006#
          8040#   8049#   8059#   8064#   8069#   8074#   8079#   8086#   8090#   8095#   8104#   8111#   8116#   8120#   8125#
          8133#   8140#   8143#   8178#   8187#   8197#   8202#   8207#   8212#   8217#   8224#   8228#   8233#   8242#   8249#
          8254#   8258#   8263#   8271#   8278#   8281#   8326#   8335#   8345#   8350#   8355#   8360#   8365#   8374#   8378#
          8383#   8397#   8402#   8406#   8411#   8424#   8434#   8441#   8444#   8488#   8497#   8507#   8512#   8517#   8522#
          8527#   8535#   8539#   8544#   8558#   8563#   8567#   8572#   8585#   8595#   8602#   8605#   8641#   8650#   8660#
          8665#   8670#   8675#   8680#   8693#   8701#   8706#   8715#   8720#   8724#   8729#   8742#   8752#   8759#   8764#
```

	8818#	8827#	8837#	8842#	8847#	8852#	8857#	8864#	8868#	8873#	8886#	8891#	8895#	8900#	8916#
	8929#	8938#	8950#	8959#	8968#	8978#	8985#	8988#	9045#	9054#	9064#	9069#	9074#	9079#	9084#
	9096#	9104#	9109#	9118#	9123#	9127#	9132#	9150#	9157#	9164#	9169#	9180#	9214#	9223#	9233#
	9238#	9243#	9248#	9253#	9265#	9273#	9278#	9287#	9292#	9296#	9301#	9320#	9327#	9334#	9339#
	9350#	9387#	9396#	9406#	9411#	9416#	9421#	9426#	9433#	9437#	9442#	9452#	9457#	9461#	9466#
	9474#	9484#	9491#	9494#	9552#	9561#	9571#	9576#	9581#	9586#	9591#	9607#	9614#	9619#	9628#
	9633#	9637#	9642#	9657#	9664#	9671#	9676#	9689#	9741#	9750#	9760#	9765#	9770#	9775#	9780#
	9792#	9800#	9805#	9814#	9819#	9823#	9828#	9838#	9845#	9852#	9857#	9872#	9904#	9913#	9923#
	9928#	9933#	9938#	9943#	9959#	9963#	9968#	9977#	9982#	9986#	9991#	10000#	10007#	10014#	10019#
	10033#	10143#	10152#	10162#	10167#	10172#	10177#	10182#	10205#	10214#	10219#	10228#	10233#	10237#	10242#
	10254#	10270#	10284#	10291#	10294#	10366#	10375#	10385#	10390#	10395#	10400#	10405#	10416#	10424#	10429#
	10447#	10452#	10455#	10466#	10473#	10483#	10521#	10530#	10540#	10545#	10550#	10555#	10560#	10566#	10570#
	10575#	10586#	10591#	10595#	10600#	10609#	10619#	10626#	10629#	10681#	10690#	10700#	10705#	10710#	10715#
	10720#	10732#	10736#	10741#	10751#	10756#	10760#	10765#	10773#	10780#	10785#	10789#	10794#	10799#	10808#
	10848#	10857#	10867#	10872#	10877#	10882#	10887#	10892#	10896#	10901#	10918#	10923#	10927#	10932#	10941#
	10951#	10958#	10961#	11020#	11029#	11039#	11044#	11049#	11054#	11059#	11068#	11075#	11080#	11094#	11099#
	11103#	11108#	11116#	11126#	11134#	11142#	11188#	11197#	11207#	11212#	11217#	11222#	11227#	11241#	11245#
	11250#	11264#	11269#	11273#	11278#	11289#	11297#	11307#	11314#	11323#	11377#	11386#	11396#	11401#	11406#
	11411#	11416#	11428#	11438#	11448#	11453#	11462#	11467#	11471#	11476#	11485#	11495#	11502#	11515#	11525#
	11585#	11594#	11604#	11609#	11614#	11619#	11624#	11634#	11647#	11652#	11661#	11666#	11670#	11675#	11687#
	11702#	11713#	11720#	11727#	11732#	11745#	11800#	11809#	11819#	11824#	11829#	11834#	11839#	11845#	11858#
	11863#	11872#	11877#	11881#	11886#	11895#	11910#	11921#	11928#	11935#	11942#	12036#	12045#	12055#	12060#
	12065#	12070#	12075#	12091#	12099#	12104#	12113#	12118#	12122#	12127#	12144#	12155#	12162#	12167#	12176#
	12181#	12190#	12196#	12201#	12216#	12227#	12236#	12241#	12248#	12253#	12262#	12267#	12271#	12276#	12295#
	12305#	12312#	12315#	12379#	12388#	12398#	12403#	12408#	12413#	12418#	12425#	12432#	12437#	12446#	12451#
	12455#	12460#	12469#	12479#	12486#	12492#	12499#	12504#	12513#	12518#	12522#	12527#	12537#	12542#	12550#
	12557#	12566#	12626#	12635#	12645#	12650#	12655#	12660#	12665#	12680#	12684#	12689#	12698#	12703#	12707#
	12712#	12720#	12725#	12751#	12756#	12763#	12766#	12790#	12796#	12812#	12817#	12825#	12830#	12846#	12851#
	12859#	12864#	12890#	12895#	12903#	12908#	12935#	12940#	12947#	12952#	12981#	12988#	12995#	13002#	13009#
	13015#														
MSWORD	1#	1416#	1468#	1477	1523#	1525	1526	1527	1528	1529	1530	1531	1532	1533	1534
	1535	1536	1537	1538	1539	1540	1541	1542	1543	1544	1545	1546	1547	1548	1549
	1550	1551	1552	1553	1554	1555	1556	1557	1558	1559	1560	1561	1562	1563	1564
	1565	1566	1567	1568	1569	1570	4718#	4891#	4911#	4922#	5034#	5035	5036	5037	5068#
	5069	5070	5071	5085#	5086	5087	5088	5137#	5138	5139	5140	5171#	5172	5173	5174
	5188#	5189	5190	5191	5205#	5206	5207	5208	5222#	5223	5224	5225	5239#	5240	5241
	5242	5283#	5284	5285	5286	5333#	5334	5335	5336	5395#	5396	5397	5398	5424#	5425
	5426	5427	5448#	5449	5450	5451	5479#	5480	5481	5482	5503#	5504	5505	5506	5528#
	5529	5530	5531	5547#	5548	5549	5550	5595#	5596	5597	5598	5635#	5636	5637	5638
	5651#	5652	5653	5654	5704#	5705	5706	5707	5726#	5727	5728	5729	5751#	5752	5753
	5754	5768#	5769	5770	5771	5805#	5806	5807	5808	5822#	5823	5824	5825	5846#	5847
	5848	5849	5859#	5860	5861	5862	6242#	6243	6244	6245	6262#	6263	6264	6265	6308#
	6309	6310	6311	6328#	6329	6330	6331	6348#	6349	6350	6351	6439#	6440	6441	6442
	6459#	6460	6461	6462	6482#	6483	6484	6485	6502#	6503	6504	6505	6543#	6544	6545
	6546	6560#	6561	6562	6563	6577#	6578	6579	6580	6669#	6670	6671	6672	6689#	6690
	6691	6692	6716#	6717	6718	6719	6736#	6737	6738	6739	6764#	6765	6766	6767	6774#
	6775	6776	6777	6843#	6844	6845	6846	6851#	6852	6853	6854	6871#	6872	6873	6874
	6892#	6893	6894	6895	6899#	6900	6901	6902	6919#	6920	6921	6922	6926#	6927	6928
	6929	6990#	6991	6992	6993	7018#	7019	7020	7021	7034#	7035	7036	7037	7049#	7050
	7051	7052	7120#	7121	7122	7123	7130#	7131	7132	7133	7166#	7167	7168	7169	7194#
	7195	7196	7197	7206#	7207	7208	7209	7233#	7234	7235	7236	7301#	7302	7303	7304
	7311#	7312	7313	7314	7333#	7334	7335	7336	7345#	7346	7347	7348	7366#	7367	7368
	7369	7373#	7374	7375	7376	7447#	7448	7449	7450	7457#	7458	7459	7460	7480#	7481
	7482	7483	7501#	7502	7503	7504	7510#	7511	7512	7513	7533#	7534	7535	7536	7540#
	7541	7542	7543	7552#	7553	7554	7555	7612#	7613	7614	7615	7622#	7623	7624	7625
	7650#	7651	7652	7653	7665#	7666	7667	7668	7674#	7675	7676	7677	7691#	7692	7693

7694	7704#	7705	7706	7707	7771#	7772	7773	7774	7781#	7782	7783	7784	7804#	7805
7806	7807	7818#	7819	7820	7821	7827#	7828	7829	7830	7846#	7847	7848	7849	7860#
7861	7862	7863	7921#	7922	7923	7924	7929#	7930	7931	7932	7951#	7952	7953	7954
7965#	7966	7967	7968	7974#	7975	7976	7977	7991#	7992	7993	7994	8059#	8060	8061
8062	8069#	8070	8071	8072	8090#	8091	8092	8093	8104#	8105	8106	8107	8111#	8112
8113	8114	8120#	8121	8122	8123	8133#	8134	8135	8136	8197#	8198	8199	8200	8207#
8208	8209	8210	8228#	8229	8230	8231	8242#	8243	8244	8245	8249#	8250	8251	8252
8258#	8259	8260	8261	8271#	8272	8273	8274	8345#	8346	8347	8348	8355#	8356	8357
8358	8378#	8379	8380	8381	8397#	8398	8399	8400	8406#	8407	8408	8409	8424#	8425
8426	8427	8434#	8435	8436	8437	8507#	8508	8509	8510	8517#	8518	8519	8520	8539#
8540	8541	8542	8558#	8559	8560	8561	8567#	8568	8569	8570	8585#	8586	8587	8588
8595#	8596	8597	8598	8660#	8661	8662	8663	8670#	8671	8672	8673	8701#	8702	8703
8704	8715#	8716	8717	8718	8724#	8725	8726	8727	8742#	8743	8744	8745	8752#	8753
8754	8755	8837#	8838	8839	8840	8847#	8848	8849	8850	8868#	8869	8870	8871	8886#
8887	8888	8889	8895#	8896	8897	8898	8916#	8917	8918	8919	8929#	8930	8931	8932
8938#	8939	8940	8941	8950#	8951	8952	8953	8959#	8960	8961	8962	8968#	8969	8970
8971	8978#	8979	8980	8981	9064#	9065	9066	9067	9074#	9075	9076	9077	9104#	9105
9106	9107	9118#	9119	9120	9121	9127#	9128	9129	9130	9150#	9151	9152	9153	9164#
9165	9166	9167	9233#	9234	9235	9236	9243#	9244	9245	9246	9273#	9274	9275	9276
9287#	9288	9289	9290	9296#	9297	9298	9299	9320#	9321	9322	9323	9334#	9335	9336
9337	9406#	9407	9408	9409	9416#	9417	9418	9419	9437#	9438	9439	9440	9452#	9453
9454	9455	9461#	9462	9463	9464	9474#	9475	9476	9477	9484#	9485	9486	9487	9571#
9572	9573	9574	9581#	9582	9583	9584	9614#	9615	9616	9617	9628#	9629	9630	9631
9637#	9638	9639	9640	9657#	9658	9659	9660	9671#	9672	9673	9674	9760#	9761	9762
9763	9770#	9771	9772	9773	9800#	9801	9802	9803	9814#	9815	9816	9817	9823#	9824
9825	9826	9838#	9839	9840	9841	9852#	9853	9854	9855	9923#	9924	9925	9926	9933#
9934	9935	9936	9963#	9964	9965	9966	9977#	9978	9979	9980	9986#	9987	9988	9989
10000#	10001	10002	10003	10014#	10015	10016	10017	10162#	10163	10164	10165	10172#	10173	10174
10175	10214#	10215	10216	10217	10228#	10229	10230	10231	10237#	10238	10239	10240	10254#	10255
10256	10257	10270#	10271	10272	10273	10284#	10285	10286	10287	10385#	10386	10387	10388	10395#
10396	10397	10398	10424#	10425	10426	10427	10447#	10448	10449	10450	10455#	10456	10457	10458
10466#	10467	10468	10469	10540#	10541	10542	10543	10550#	10551	10552	10553	10570#	10571	10572
10573	10586#	10587	10588	10589	10595#	10596	10597	10598	10609#	10610	10611	10612	10619#	10620
10621	10622	10700#	10701	10702	10703	10710#	10711	10712	10713	10736#	10737	10738	10739	10751#
10752	10753	10754	10760#	10761	10762	10763	10773#	10774	10775	10776	10789#	10790	10791	10792
10867#	10868	10869	10870	10877#	10878	10879	10880	10896#	10897	10898	10899	10918#	10919	10920
10921	10927#	10928	10929	10930	10941#	10942	10943	10944	10951#	10952	10953	10954	11039#	11040
11041	11042	11049#	11050	11051	11052	11075#	11076	11077	11078	11094#	11095	11096	11097	11103#
11104	11105	11106	11116#	11117	11118	11119	11126#	11127	11128	11129	11207#	11208	11209	11210
11217#	11218	11219	11220	11245#	11246	11247	11248	11264#	11265	11266	11267	11273#	11274	11275
11276	11289#	11290	11291	11292	11297#	11298	11299	11300	11307#	11308	11309	11310	11396#	11397
11398	11399	11406#	11407	11408	11409	11448#	11449	11450	11451	11462#	11463	11464	11465	11471#
11472	11473	11474	11485#	11486	11487	11488	11495#	11496	11497	11498	11604#	11605	11606	11607
11614#	11615	11616	11617	11647#	11648	11649	11650	11661#	11662	11663	11664	11670#	11671	11672
11673	11687#	11688	11689	11690	11702#	11703	11704	11705	11713#	11714	11715	11716	11727#	11728
11729	11730	11819#	11820	11821	11822	11829#	11830	11831	11832	11858#	11859	11860	11861	11872#
11873	11874	11875	11881#	11882	11883	11884	11895#	11896	11897	11898	11910#	11911	11912	11913
11921#	11922	11923	11924	11935#	11936	11937	11938	12055#	12056	12057	12058	12065#	12066	12067
12068	12099#	12100	12101	12102	12113#	12114	12115	12116	12122#	12123	12124	12125	12144#	12145
12146	12147	12155#	12156	12157	12158	12176#	12177	12178	12179	12190#	12191	12192	12193	12196#
12197	12198	12199	12216#	12217	12218	12219	12227#	12228	12229	12230	12248#	12249	12250	12251
12262#	12263	12264	12265	12271#	12272	12273	12274	12295#	12296	12297	12298	12305#	12306	12307
12308	12398#	12399	12400	12401	12408#	12409	12410	12411	12432#	12433	12434	12435	12446#	12447
12448	12449	12455#	12456	12457	12458	12469#	12470	12471	12472	12479#	12480	12481	12482	12499#
12500	12501	12502	12513#	12514	12515	12516	12522#	12523	12524	12525	12537#	12538	12539	12540
12550#	12551	12552	12553	12645#	12646	12647	12648	12655#	12656	12657	12658	12684#	12685	12686

```
          12687   12698#  12699   12700   12701   12707#  12708   12709   12710   12720#  12721   12722   12723   12751#  12752
          12753   12754   12790#  12812#  12813   12814   12815   12825#  12826   12827   12828   12846#  12847   12848   12849
          12859#  12860   12861   12862   12890#  12891   12892   12893   12903#  12904   12905   12906   12935#  12936   12937
          12938   12947#  12948   12949   12950   13142#  13147#
MSXFER       1#    1416#
OPEN         1#    1416#
POINTE       1#    1416#   1426
PRINTB       1#    1416#   2916    2929    2945    2952    2966    2979    2992    3009    3015    3028    3045    3058    3064
          3079     3085    3092    3105    3118    3130    3143    3155    3161    3167    3180    3193    3200    3213    3227
          3242     3249    3256    3270    3277    3284    3298    3312    3319    3327    3341    3347    3354    3368    3374
          3381     3396    3404    3414    3430    3442    3448    3457    3464    3482    3489    3495    3501    3508    3515
          3528     3534    3547    3553    3567    3573    3587    3595    3602    3615    3621    3634    3640    3647    3654
          3661     3675    3681    3689    3702    3715    3722    3735    3742    3748    3762    3774    3786    3798    3810
          3816     3823    3837    3843    3857    3864    3871    3878    3885    3899    3913    3921    3929    3937    3950
          3956     3962    3974   12976   12983   12990   12997   13004
PRINTF       1#    1416#   4385    4538    4553    4574    4640    4655    4668    4683    4696    4795    4881
PRINTS       1#    1416#
PRINTX       1#    1416#   4421
READBU       1#    1416#
READEF       1#    1416#   4753    4758    4773    4778
RFLAGS       1#    1416#
SETPRI       1#    1416#   4232    4246    7007    7180
SETVEC       1#    1416#   4808    5015    5118    5270    5318    6971    7137
SLASH        1#    1416#
STARS        1#    1416#
SVC          1#    1416#
XFER         1#    1416#   4718#   4891#   4911#   4922#  12790#
XFERF        1#    1416#
XFERT        1#    1416#


. ABS.   000000     000
         000000     001
UNAREP   053566     002
MICRA    000000     003
MICRB    000000     004
MICRC    000000     005
MICRD    000000     006
MICRE    000000     007
MICRF    000000     010
MICRG    000000     011
NOMORE   000000     012


 ERRORS DETECTED:  0
% DEFAULT GLOBALS GENERATED:  15

 CZUAAB.OBJ,CZUAAB.LST/CR/SOL/NL:TOC=SVC34R.P11,CZUAAB.MAC
 RUN-TIME: 72 86 10 SECONDS
 RUN-TIME RATIO: 215/168=1.2
 CORE USED:  31K  (61 PAGES)
```

```

                                 .TITLE  MICROA - MICROCODE MODULE A

        000000'                  .CSECT MICRA

                                 .SBTTL   REGISTER DEFINITIONS USED BY THE T11

        021000                   IPCSR0  =       21000   ;INTERNAL PCSR0 ADDRESS
        021002                   DMACSR  =       21002   ;DMA ENGINE CONTROL STATUS REGISTER
        021004                   DMAT0   =       21004   ;DMA ENGINE TO ADDRESS REGISTER #0
        021006                   DMAT1   =       21006   ;DMA ENGINE TO ADDRESS REGISTER #1
        021010                   MDMA0   =       21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
        021012                   MDMA1   =       21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
        021014                   MDMAR0  =       21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
        021016                   MDMAR1  =       21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
        021020                   IPCSR1  =       21020   ;INTERNAL PCSR1 ADDRESS
        021022                   DMAF    =       21022   ;DMA ENGINE FROM ADDRESS REGISTER
        021024                   DMAWC   =       21024   ;DMA ENGINE WORD COUNT REGISTER
        021026                   MDMAW0  =       21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
        021030                   LTAC    =       21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
        021032                   LFRBUF  =       21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
        021034                   CLRFIF  =       21034   ;CLEAR FIFO
        021036                   MDMAW1  =       21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
        021040                   PCSRSW  =       21040   ;SWITCH PACK REGISTER
        021042                   MDMSR   =       21042   ;MICROCPU DMA STATUS REGISTER
        021044                   LRBUF   =       21044   ;LINK RECIEVE BUFFER COMPLETED
        021060                   PHYAD0  =       21060   ;PHYSICAL ADDRESS ROM BYTE 0
        021062                   PHYAD1  =       21062   ;PHYSICAL ADDRESS ROM BYTE 1
        021064                   PHYAD2  =       21064   ;PHYSCIAL ADDRESS ROM BYTE 2
        021066                   PHYAD3  =       21066   ;PHYSICAL ADDRESS ROM BYTE 3
        021070                   PHYAD4  =       21070   ;PHYSICAL ADDRESS ROM BYTE 4
        021072                   PHYAD5  =       21072   ;PHYSICAL ADDRESS ROM BYTE 5

                                 .SBTTL   OTHER DEFINITIONS USED BY THE MICROCODE

        100000                   BIT15   =       100000
        040000                   BIT14   =       40000
        020000                   BIT13   =       20000
        010000                   BIT12   =       10000
        004000                   BIT11   =       4000
        002000                   BIT10   =       2000
        001000                   BIT9    =       1000
        000400                   BIT8    =       400
        000200                   BIT7    =       200
        000100                   BIT6    =       100
        000040                   BIT5    =       40
        000020                   BIT4    =       20
        000010                   BIT3    =       10
        000004                   BIT2    =       4
        000002                   BIT1    =       2
        000001                   BIT0    =       1
                                 ;
        012400                   LASFTP  =       BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
        000340                   PRI07   =       340
        000300                   PRI06   =       300
        000240                   PRI05   =       240
```

```
        000200          PRI04   =       200
        000140          PRI03   =       140
        000100          PRI02   =       100
        000040          PRI01   =       40
        000000          PRI00   =       0
                        ;
                        ;PCSR0 - PORT CONTROL STATUS REGISTER 0
                        ;
        100000          SERI    =       BIT15
        040000          PCEI    =       BIT14
        020000          RXI     =       BIT13
        010000          TXI     =       BIT12
        004000          DNI     =       BIT11
        002000          RCEI    =       BIT10
        000400          FATI    =       BIT8
                        ;
        000134          SANVEC= 134             ;VECTOR ADDRESS FOR THE SANITY TIMER
        000150          NXMVEC= 150             ;VECTOR ADDRESS FOR THE NON-EXISTANT MEMORY TIMOUT
        000064          CSRVEC= 64              ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
        000114          DMAVEC= 114             ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
        000140          PARVEC= 140             ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
        000150          NXMVEC= 150             ;VECTOR ADDRESS FOR NON-EXISTANT UNIBUS ADDRESS
        001000          STACK=  1000            ;STACK LOCATION
        000001          INMON=  1               ;IN MICROMONITOR STATE  *** FOR ASSEMBLY WITH MACY11
        000002          INTST=  2               ;IN A TEST STATE        *** THESE THREE VARIABLES MUST
        000003          INERR=  3               ;IN ERROR STATE         *** BE CHANGED TO LOCAL(I.E.=)
        000001          CSRFLG= BIT0            ;CSR WRITE INTERRUPT OCCURED
        000002          ERRFLG= BIT1            ;UNEXPECTED ERROR OCCURED
        000004          PARFLG= BIT2            ;PARITY ERROR OCCURED
        000010          NXMFLG= BIT3            ;NON-EXISTANT MEMORY ERROR OCCURRED
        020000          SIZ4K=  20000           ;4K WORDS
        040000          SIZ8K=  SIZ4K*2         ;8K WORDS
        020000          WCSSIZ= SIZ4K           ;SIZE OF WRITEABLE CONTROL STORE
        020000          IOSIZ=  SIZ4K           ;SIZE OF I/O PAGE
        040000          ROMSIZ= SIZ8K           ;SIZE OF ROM
        077774          LINSIZ= SIZ8K*2-4       ;SIZE OF LINK MEMORY
        000000          WCSADR= 0               ;BASE ADDRESS OF WCS
        020000          IOADR=  WCSADR+WCSSIZ   ;BASE ADDRESS OF I/O PAGE
        040000          ROMADR= IOADR+IOSIZ     ;BASE ADDRESS OF ROM
        100000          LINADR= ROMADR+ROMSIZ   ;BASE ADDRESS OF LINK MEMORY
        000000          DATERR= 0               ;FLAG INDICATING DATA ERROR OCCURRED
        000001          ADRERR= 1               ;FLAG INDICATING ADDRESS ERROR OCCURRED
        000002          PARERR= 2               ;FLAG INDICATING PARITY ERROR OCCURRED
        177774          MODREG= LINADR+LINSIZ   ;LINK MODE REGISTER
        177774          ADDREG= MODREG          ;LINK STATION ADDRESS RAM REGISTER
        177776          CMDREG= MODREG+2        ;LINK COMMAND REGISTER
```

```


                                .SBTTL  A_MODULE MICROCODE

                                ;*********************************************************************
                                ;
                                ;THIS MODULE CONTAINS MICROCODE THAT IS USED FOR THE LOAD AND START
                                ;FUNCTION TEST
                                ;
                                ;*********************************************************************


000000' 106427  000340          MICROA::MTPS    #PRI07                  ;DISABLE INTERRUPTS
000004' 012737  012402  021020          MOV     #LASFTP!INTST,@#IPCSR1  ;SET TEST PATTERN AND IN TEST STATE BITS
000012' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;SET DONE BIT
000020' 000777                          BR      .                       ;HANG HERE UNTIL HOST RESETS US

000022' 000024                  MICASZ::MICASZ-MICROA+2                 ;SIZE OF MICROCODE MODULE A
        000001                  .END
```

```
ADDREG= 177774                  101#
ADRERR= 000001                   98#
BIT0  = 000001                   51#     83
BIT1  = 000002                   50#     84
BIT10 = 002000                   41#     53      70
BIT11 = 004000                   40#     69
BIT12 = 010000                   39#     53      68
BIT13 = 020000                   38#     67
BIT14 = 040000                   37#     66
BIT15 = 100000                   36#     65
BIT2  = 000004                   49#     85
BIT3  = 000010                   48#     86
BIT4  = 000020                   47#
BIT5  = 000040                   46#
BIT6  = 000100                   45#
BIT7  = 000200                   44#
BIT8  = 000400                   43#     53      71
BIT9  = 001000                   42#
CLRFIF= 021034                   22#
CMDREG= 177776                  102#
CSRFLG= 000001                   83#
CSRVEC= 000064                   75#
DATERR= 000000                   97#
DMACSR= 021002                    9#
DMAF  = 021022                   17#
DMAT0 = 021004                   10#
DMAT1 = 021006                   11#
DMAVEC= 000114                   76#
DMAWC = 021024                   18#
DNI   = 004000                   69#    118
ERRFLG= 000002                   84#
FATI  = 000400                   71#
INERR = 000003                   82#
INMON = 000001                   80#
INTST = 000002                   81#    117
IOADR = 020000                   94#     95
IOSIZ = 020000                   90#     95
IPCSR0= 021000                    8#    118*
IPCSR1= 021020                   16#    117*
LASFTP= 012400                   53#    117
LFRBUF= 021032                   21#
LINADR= 100000                   96#    100
LINSIZ= 077774                   92#    100
LRBUF = 021044                   26#
LTAC  = 021030                   20#
MDMAR0= 021014                   14#
MDMAR1= 021016                   15#
MDMAW0= 021026                   19#
MDMAW1= 021036                   23#
MDMA0 = 021010                   12#
MDMA1 = 021012                   13#
MDMSR = 021042                   25#
MICASZ  000022RG      002       121#
MICROA  000000RG      002       116#    121
MODREG= 177774                  100#    101     102
NXMFLG= 000010                   86#
```

```
NXMVEC= 000150          74#     78#
PARERR= 000002          99#
PARFLG= 000004          85#
PARVEC= 000140          77#
PCEI   = 040000         66#
PCSRSW= 021040          24#
PHYAD0= 021060          27#
PHYAD1= 021062          28#
PHYAD2= 021064          29#
PHYAD3= 021066          30#
PHYAD4= 021070          31#
PHYAD5= 021072          32#
PRI00 = 000000          61#
PRI01 = 000040          60#
PRI02 = 000100          59#
PRI03 = 000140          58#
PRI04 = 000200          57#
PRI05 = 000240          56#
PRI06 = 000300          55#
PRI07 = 000340          54#    116*
RCEI   = 002000         70#
ROMADR= 040000          95#     96
ROMSIZ= 040000          91#     96
RXI    = 020000         67#
SANVEC= 000134          73#
SERI   = 100000         65#
SIZ4K = 020000          87#     88      89      90
SIZ8K = 040000          88#     91      92
STACK = 001000          79#
TXI    = 010000         68#
WCSADR= 000000          93#     94
WCSSIZ= 020000          89#     94
.      = 000024R    002    119
```

```
BCOMPL     1#
BERROR     1#
BGNAU      1#
BGNAUT     1#
BGNCLN     1#
BGNDU      1#
BGNHRD     1#
BGNHW      1#
BGNINI     1#
BGNMOD     1#
BGNMSG     1#
BGNPRO     1#
BGNPTA     1#
BGNRPT     1#
BGNSEG     1#
BGNSET     1#
BGNSFT     1#
BGNSRV     1#
BGNSUB     1#
BGNSW      1#
BGNTST     1#
BNCOMP     1#
BNERRO     1#
BREAK      1#
BRESET     1#
CKLOOP     1#
CLOCK      1#
CLOSE      1#
CLRVEC     1#
COMMEN     1#
DELAY      1#
DESCRI     1#
DEVTYP     1#
DISPAT     1#
DISPLA     1#
DOCLN      1#
DODU       1#
DORPT      1#
ENDAU      1#
ENDAUT     1#
ENDCLN     1#
ENDCOM     1#
ENDDU      1#
ENDHRD     1#
ENDHW      1#
ENDINI     1#
ENDMOD     1#
ENDMSG     1#
ENDPRO     1#
ENDPTA     1#
ENDRPT     1#
ENDSEG     1#
ENDSET     1#
ENDSFT     1#
ENDSRV     1#
ENDSUB     1#
```

```
ENDSY      1#
ENDTST     1#
EQUALS     1#
ERRDF      1#
ERRHRD     1#
ERROR      1#
ERRSF      1#
ERRSOF     1#
ERRTBL     1#
ESCAPE     1#
EXIT       1#
FEQUAL     1#
GETBYT     1#
GETPRI     1#
GETWOR     1#
GMANIA     1#
GMANID     1#
GMANIL     1#
GPHARD     1#
GPRMA      1#
GPRMD      1#
GPRML      1#
HEADER     1#
INLOOP     1#
IOSETU     1#
IOSTAR     1#
KT11       1#
LASTAD     1#
MANUAL     1#
MEMORY     1#
MSBYTE     1#
MSCHEC     1#
MSCNTO     1#
MSCOUN     1#
MSDATA     1#
MSDECR     1#
MSDEFA     1#
MSENDE     1#
MSERRI     1#
MSESCA     1#
MSESCS     1#
MSEXCP     1#
MSEXIT     1#
MSEXSE     1#
MSEXTJ     1#
MSGEN      1#
MSGENB     1#
MSGETS     1#
MSGETT     1#
MSGNGB     1#
MSGNIN     1#
MSGNLS     1#
MSGNSU     1#
MSGNTA     1#
MSGNTE     1#
MSHAPT     1#
```

```
MSHNAP     1#
MSINCR     1#
MSIOSE     1#
MSLDRO     1#
MSMASK     1#
MSMCHI     1#
MSMCLO     1#
MSMSK1     1#
MSPOP      1#
MSPRIN     1#
MSPUSH     1#
MSPUT      1#
MSPUT1     1#
MSRADI     1#
MSRBRO     1#
MSRNRO     1#
MSSETS     1#
MSSTAR     1#
MSSVC      1#
MSTLAB     1#
MSTSTL     1#
MSWORD     1#
MSXFER     1#
OPEN       1#
POINTE     1#
PRINTB     1#
PRINTF     1#
PRINTS     1#
PRINTX     1#
READBU     1#
READEF     1#
RFLAGS     1#
SETPRI     1#
SETVEC     1#
SLASH      1#
STARS      1#
SVC        1#
XFER       1#
XFERF      1#
XFERT      1#


. ABS.  000000      000
        000000      001
MICRA   000024      002


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROA.OBJ,MICROA.LST/CR/SOL/NL:TOC=SVC34R.P11,MICROA.MAC
 RUN-TIME: 2 1 .2 SECONDS
 RUN-TIME RATIO: 39/4=8.8
 CORE USED:  31K  (61 PAGES)
```

```
                                .TITLE  MICROB - MICROCODE MODULE B

        000000'                 .CSECT MICRB

                                .SBTTL   REGISTER DEFINITIONS USED BY THE T11

        021000                  IPCSR0  =       21000   ;INTERNAL PCSR0 ADDRESS
        021002                  DMACSR  =       21002   ;DMA ENGINE CONTROL STATUS REGISTER
        021004                  DMAT0   =       21004   ;DMA ENGINE TO ADDRESS REGISTER #0
        021006                  DMAT1   =       21006   ;DMA ENGINE TO ADDRESS REGISTER #1
        021010                  MDMA0   =       21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
        021012                  MDMA1   =       21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
        021014                  MDMAR0  =       21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
        021016                  MDMAR1  =       21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
        021020                  IPCSR1  =       21020   ;INTERNAL PCSR1 ADDRESS
        021022                  DMAF    =       21022   ;DMA ENGINE FROM ADDRESS REGISTER
        021024                  DMAWC   =       21024   ;DMA ENGINE WORD COUNT REGISTER
        021026                  MDMAW0  =       21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
        021030                  LTAC    =       21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
        021032                  LFRBUF  =       21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
        021034                  CLRFIF  =       21034   ;CLEAR FIFO
        021036                  MDMAW1  =       21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
        021040                  PCSRSW  =       21040   ;SWITCH PACK REGISTER
        021042                  MDMSR   =       21042   ;MICROCPU DMA STATUS REGISTER
        021044                  LRBUF   =       21044   ;LINK RECIEVE BUFFER COMPLETED
        021060                  PHYAD0  =       21060   ;PHYSICAL ADDRESS ROM BYTE 0
        021062                  PHYAD1  =       21062   ;PHYSICAL ADDRESS ROM BYTE 1
        021064                  PHYAD2  =       21064   ;PHYSCIAL ADDRESS ROM BYTE 2
        021066                  PHYAD3  =       21066   ;PHYSICAL ADDRESS ROM BYTE 3
        021070                  PHYAD4  =       21070   ;PHYSICAL ADDRESS ROM BYTE 4
        021072                  PHYAD5  =       21072   ;PHYSICAL ADDRESS ROM BYTE 5

                                .SBTTL   OTHER DEFINITIONS USED BY THE MICROCODE

        100000                  BIT15   =       100000
        040000                  BIT14   =       40000
        020000                  BIT13   =       20000
        010000                  BIT12   =       10000
        004000                  BIT11   =       4000
        002000                  BIT10   =       2000
        001000                  BIT9    =       1000
        000400                  BIT8    =       400
        000200                  BIT7    =       200
        000100                  BIT6    =       100
        000040                  BIT5    =       40
        000020                  BIT4    =       20
        000010                  BIT3    =       10
        000004                  BIT2    =       4
        000002                  BIT1    =       2
        000001                  BIT0    =       1
                                ;
        012400                  LASFTP  =       BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
        000340                  PRI07   =       340
        000300                  PRI06   =       300
        000240                  PRI05   =       240
```

```
        000200          PRI04   =       200
        000140          PRI03   =       140
        000100          PRI02   =       100
        000040          PRI01   =       40
        000000          PRI00   =       0
                        ;
                        ;PCSR0 - PORT CONTROL STATUS REGISTER 0
                        ;
        100000          SERI    =       BIT15
        040000          PCEI    =       BIT14
        020000          RXI     =       BIT13
        010000          TXI     =       BIT12
        004000          DNI     =       BIT11
        002000          RCEI    =       BIT10
        000400          FATI    =       BIT8
                        ;
        000134          SANVEC= 134                 ;VECTOR ADDRESS FOR THE SANITY TIMER
        000064          CSRVEC= 64                  ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
        000114          DMAVEC= 114                 ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
        000140          PARVEC= 140                 ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
        001000          STACK=  1000                ;STACK LOCATION
        000001          INMON=  1                   ;IN MICROMONITOR STATE
        000002          INTST=  2                   ;IN A TEST STATE
        000003          INERR=  3                   ;IN ERROR STATE
        000001          CSRFLG= BIT0                ;CSR WRITE INTERRUPT OCCURED
        000002          ERRFLG= BIT1                ;UNEXPECTED ERROR OCCURED
        000004          PARFLG= BIT2                ;PARITY ERROR OCCURED
        000010          NXMFLG= BIT3                ;NON-EXISTANT MEMORY ERROR OCCURRED
        000020          NPRFLG= BIT4                ;NPR TIMEOUT OCCURRED
        100000          NPRERR= BIT15               ;PCSR0 FLAG INDICATING NPR ERROR OCCURRED
        040000          NXMERR= BIT14               ;PCSR0 FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
        020000          UNIERR= BIT13               ;PCSR0 FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
        010000          PARERR= BIT12               ;PCSR0 FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
        020000          SIZ4K=  20000               ;4K WORDS
        040000          SIZ8K=  SIZ4K*2             ;8K WORDS
        020000          WCSSIZ= SIZ4K               ;SIZE OF WRITEABLE CONTROL STORE
        020000          IOSIZ=  SIZ4K               ;SIZE OF I/O PAGE
        040000          ROMSIZ= SIZ8K               ;SIZE OF ROM
        077774          LINSIZ= SIZ8K*2-4                   ;SIZE OF LINK MEMORY
        000000          WCSADR= 0                   ;BASE ADDRESS OF WCS
        020000          IOADR=  WCSADR+WCSSIZ       ;BASE ADDRESS OF I/O PAGE
        040000          ROMADR= IOADR+IOSIZ         ;BASE ADDRESS OF ROM
        100000          LINADR= ROMADR+ROMSIZ       ;BASE ADDRESS OF LINK MEMORY
        000000          DATERR= 0                   ;FLAG INDICATING DATA ERROR OCCURRED
        000001          ADRERR= 1                   ;FLAG INDICATING ADDRESS ERROR OCCURRED
        177774          MODREG= LINADR+LINSIZ       ;LINK MODE REGISTER
        177774          ADDREG= MODREG              ;LINK STATION ADDRESS RAM REGISTER
        177776          CMDREG= MODREG+2            ;LINK COMMAND REGISTER
```

```

                                        .SBTTL  B_MODULE MICROCODE

                                        ;***********************************************************************
                                        ;
                                        ;THIS MODULE CONTAINS A MICROMONITOR AND MICROCODE FOR:
                                        ;
                                        ;       1-WCS MEMORY TEST
                                        ;
                                        ;***********************************************************************


000000' 106427  000340                  MICROB::MTPS    #PRI07                  ;DISABLE INTERRUPTS
000004' 012737  000000  177776                  MOV     #0,@#CMDREG             ;TURN OFF THE LINK
000012' 012706  001000                          MOV     #STACK,SP               ;SETUP STACK
000016' 012767  000001  000510                  MOV     #INMON,PCSR1            ;TELL HOST WE ARE IN MICROMONITOR
000024' 016737  000504  021020                  MOV     PCSR1,@#IPCSR1
000032' 012737  004000  021000                  MOV     #DNI,@#IPCSR0           ;TELL HOST THE LOAD AND START FINISHED
000040' 010700                                  MOV     PC,R0                   ;GET ADDRESS OF UNEXPECTED ERROR...
000042' 062700  000344                          ADD     #ERRINT-.,R0            ;HANDLER
000046' 005001                                  CLR     R1                      ;FILL ALL UNUSED VECTORS WITH TRAP...
000050' 010021                          10$:    MOV     R0,(R1)+                ;HANDLER
000052' 012721  000340                          MOV     #PRI07,(R1)+
000056' 020127  001000                          CMP     R1,#1000
000062' 002772                                  BLT     10$

000064' 010700                                  MOV     PC,R0                   ;SETUP PARITY TRAP VECTOR
000066' 062700  000422                          ADD     #PARINT-.,R0
000072' 010037  000140                          MOV     R0,@#PARVEC
000076' 012737  000340  000142                  MOV     #PRI07,@#PARVEC+2

000104' 010700                                  MOV     PC,R0                   ;SETUP DMA INTERRUPT VECTOR
000106' 062700  000324                          ADD     #DMAINT-.,R0
000112' 010037  000114                          MOV     R0,@#DMAVEC
000116' 012737  000340  000116                  MOV     #PRI07,@#DMAVEC+2

000124' 010700                                  MOV     PC,R0                   ;SETUP CSR WRITE VECTOR
000126' 062700  000250                          ADD     #CSRWRT-.,R0
000132' 010037  000064                          MOV     R0,@#CSRVEC
000136' 012737  000200  000066                  MOV     #PRI04,@#CSRVEC+2

000144' 010700                                  MOV     PC,R0                   ;SETUP SANTITY TIMER VECTOR
000146' 062700  000256                          ADD     #TIMINT-.,R0
000152' 010037  000134                          MOV     R0,@#SANVEC
000156' 012737  000240  000136                  MOV     #PRI05,@#SANVEC+2

000164' 013700  021040                          MOV     @#PCSRSW,R0             ;GET SWITCH PACK BITS
000170' 052700  176000                          BIS     #176000,R0              ;MAP THEM INTO HOST I/O PAGE
000174' 006300                                  ASL     R0                      ;SHIFT OVER TO POSITION CORRECTLY
000176' 006300                                  ASL     R0
000200' 006300                                  ASL     R0
000202' 062700  000004                          ADD     #4,R0                   ;PCSR2 IS PCSR0+4
000206' 010067  000324                          MOV     R0,IPCSR2               ;SAVE PCSR2 ADDRESS
000212' 012767  000003  000320                  MOV     #3,IPCSR2+2             ;HIGH ORDER BITS 17:16
000220' 005067  000304                          CLR     FLG2                    ;INITIALIZE FLAG WORD
000224' 106427  000000                  15$:    MTPS    #PRI00                  ;ALLOW INTERRUPTS
```

```
000230' 005767  000274          20$:    TST     FLG2                    ;WAIT FOR A COMMAND FROM HOST
000234' 001775                          BEQ     20$
000236' 106427  000340                  MTPS    #PR107                  ;RAISE CPU PRIORITY TO SERVICE COMMAND
000242' 032767  000001  000260          BIT     #CSRFLG,FLG2            ;DID HOST GIVE US A COMMAND?
000250' 001001                          BNE     30$                     ;YES
000252' 000777                          BR      .                       ;NO, ERROR SO JUST SIT HERE...
                                                                        ;FOR LACK OF ANYTHING BETTER TO DO
000254' 013700  021000          30$:    MOV     @#IPCSR0,R0             ;GET WHAT HOST WROTE TO PCSR0
000260' 042700  177760                  BIC     #177760,R0              ;STRIP ALL BUT COMMAND BITS
000264' 001004                          BNE     35$                     ;WAS IT THE CLEAR FUNCTION?
000266' 012737  000001  021020          MOV     #INMON,@#IPCSR1         ;YES, CLEAR SELF TEST BITS
000274' 000432                          BR      50$
000276' 022700  000017          35$:    CMP     #17,R0                  ;RETURN TO OPERATIONAL MICROCODE?
000302' 001432                          BEQ     60$                     ;YES
000304' 162700  000001                  SUB     #1,R0                   ;WILL FORM A TABLE INDEX WITH THIS
000310' 010701                          MOV     PC,R1                   ;GET ADDRESS OF OUR COMMAND TABLE
000312' 062701  000214                  ADD     #TBLB-.,R1
000316' 006300                          ASL     R0                      ;MAKE COMMAND A BYTE OFFSET
000320' 060001                          ADD     R0,R1                   ;USE IT TO INDEX INTO COMMAND TABLE
000322' 061101                          ADD     (R1),R1                 ;R1 NOW HAS COMMAND ROUTINE ADDRESS
000324' 004711                          JSR     PC,(R1)                 ;EXECUTE AS COMMANDED FROM HOST
000326' 103404                          BCS     40$                     ;ERROR OCCURRED
000330' 112767  000001  000176          MOVB    #INMON,PCSR1            ;INDICATE TO HOST WE ARE BACK IN...
000336' 000403                          BR      45$                     ;MICROMONITR
000340' 112767  000003  000166  40$:    MOVB    #INERR,PCSR1            ;INDICATE TO HOST ERROR OCCURRED
000346' 016737  000162  021020  45$:    MOV     PCSR1,@#IPCSR1
000354' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST THIS MICROTEST FINISHED
000362' 005067  000142          50$:    CLR     FLG2                    ;RESET FLAG WORD
000366' 000716                          BR      15$                     ;GO WAIT FOR ANOTHER COMMAND
000370' 005000                  60$:    CLR     R0                      ;FAKE SUCCESSFULL SELF TEST DONE
000372' 000137  040006                  JMP     @#40006                 ;START OPERATIONAL MICROCODE
000376' 052767  000001  000124  CSRWRT: BIS     #CSRFLG,FLG2            ;INDICATE A CSR WRITE INTERRUPT OCCURED
000404' 000002                          RTI
000406' 052767  000002  000114  ERRINT: BIS     #ERRFLG,FLG2            ;INDICATE A UNEXPECTED INTERRUPT OCCURED
000414' 012737  020000  021000          MOV     #UNIERR,@#IPCSR0        ;TELL HOST AN UNEXPECTED INTERRUPT
                                                                        ;HAPPENED
000422' 000777                          BR      .                       ;JUST SIT HERE AND SPIN WHEELS
000424' 005267  000102          TIMINT: INC     SANTIM                  ;COUNT TICKS AS THEY OCCUR
000430' 000002                          RTI
000432' 013767  021002  000106  DMAINT: MOV     @#DMACSR,DMDONE         ;GET DMA STATUS
000440' 032767  040000  000100          BIT     #BIT14,DMDONE           ;DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
000446' 001404                          BEQ     10$                     ;NO
000450' 012737  040000  021000          MOV     #NXMERR,@#IPCSR0        ;YES, TELL HOST A NON-EXISTANT MEMORY
                                                                        ;LOCATION WAS ADDRESSED
000456' 000407                          BR      20$
000460' 032767  100000  000060  10$:    BIT     #BIT15,DMDONE           ;DID A NPR TIMEOUT OCCUR?
000466' 001407                          BEQ     30$                     ;NO
000470' 012737  100000  021000          MOV     #NPRERR,@#IPCSR0        ;TELL HOST NPR TIMEOUT HAPPENED
```

```
000476' 012737  100000  021002  20$:     MOV     #BIT15,@#DMACSR          ;CLEAR THE INTERRUPT IN THE DMA ENGINE
000504' 000777                           BR      .                        ;SIT HERE AND SPIN WHEELS
000506' 000002                  30$:     RTI

000510' 052767  000004  000012  PARINT:  BIS     #PARFLG,FLG2             ;SET PARITY ERROR OCCURRED
000516' 012737  010000  021000           MOV     #PARERR,@#IPCSR0         ;TELL HOST A LINK MEMORY PARITY ERROR
                                                                          ;OCCURRED
000524' 000777                           BR      .                        ;WAIT FOR HOST TO TIME OUT

000526' 000024                  TBLB:    .WORD   MICB1-.                  ;WCS MEMORY TEST

000530' 000000                  FLG2:    .WORD   0                        ;FLAG WORD
000532' 000000                  SANTIM:  .WORD   0                        ;COUNT FOR SANITY TIMER
000534' 000000                  PCSR1:   .WORD   0                        ;COPY OF PCSR1(BECAUSE WE CAN'T DO BYTE)
000536' 000000  000000          IPCSR2:  .WORD   0,0                      ;ADDRESS IN HOST MEMORY FOR PCSR2
000542' 000000  000000          PCBADR:  .WORD   0,0                      ;ADDRESS IN HOST MEMORY FOR PCB
000546' 000000                  DMDONE:  .WORD   0                        ;DMA INTERRUPT HAPPENED
000550' 000377                  BAKPAT:  .WORD   377                      ;BACKGROUND PATTERN FOR MEMORY TESTS
```

```

                                          .SBTTL  MODULE B, MICROTEST #1

                                          ;********************************************************************8
                                          ;
                                          ;THIS MICROTEST CHECKS THE WCS MEMORY. IT DOES SO BY RUNNING A SERIES
                                          ;OF MICROSUBTESTS ON THE MEMORY BETWEEN 0 AND 2K(000000-10000) AND THEN
                                          ;RELOCATES ITSELF TO THE LOWER HALF OF WCS AND TESTS THE UPPER HALF.
                                          ;
                                          ;THE SUBTESTS ARE:
                                          ;       A-ACCESS TEST
                                          ;       B-ADDRESS SHIFT TEST
                                          ;       C-DATA LATCH TEST
                                          ;       D-ADDRESS BIT SHIFT #1
                                          ;       E-ADDRESS BIT SHIFT #2
                                          ;       F-MARCH TEST
                                          ;
                                          ;********************************************************************8

000552' 112767  000002  177754  MICB1:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
000560' 016737  177750  021020          MOV     PCSR1,@#IPCSR1
000566' 005067  000320                  CLR     BOTH                    ;START TESTING BOTTOM HALF FIRST
000572' 012767  001000  000304          MOV     #1000,LOWLIM            ;SET LOW LIMIT JUST ABOVE VECTORS
000600' 012767  010000  000300          MOV     #WCSSIZ/2,HILIM         ;HIGH LIMIT IS HALF WAY INTO WCS

000606' 010700                  5$:     MOV     PC,R0                   ;GET ADDRESS OF SUBTEST LIST
000610' 062700  000256                  ADD     #B1STBL-.,R0

000614' 005067  177710          10$:    CLR     FLG2                    ;CLEAR THE FLAG WORD
000620' 011001                          MOV     (R0),R1                 ;GET OFFSET FROM ENTRY TO SUBTEST
000622' 001437                          BEQ     30$                     ;NO MORE ENTRIES
000624' 060001                          ADD     R0,R1                   ;CALC SUBTEST ADDRESS
000626' 004711                          JSR     PC,(R1)                 ;GO EXECUTE SUBTEST
000630' 103403                          BCS     20$                     ;ERROR OCCURRED IN SUBTEST
000632' 062700  000002                  ADD     #2,R0                   ;POINT TO NEXT SUBTEST IN LIST
000636' 000766                          BR      10$

000640' 016737  177672  021010  20$:    MOV     IPCSR2,@#MDMA0          ;GET CONTENTS OF HOST'S PCSR2 AND PCSR3
000646' 016737  177666  021012          MOV     IPCSR2+2,@#MDMA1
000654' 013700  021014                  MOV     @#MDMAR0,R0             ;R0=CONTENTS OF PCSR2
000660' 013702  021014                  MOV     @#MDMAR0,R2             ;R2=CONTENTS OF PCSR3
000664' 010037  021010                  MOV     R0,@#MDMA0              ;POINT TO PCBB+0
000670' 010237  021012                  MOV     R2,@#MDMA1
000674' 016737  000210  021026          MOV     SUBNUM,@#MDMAW0         ;LOAD PCBB+0 WITH SUBTEST #...
                                                                        ;AND PCBB+1 WITH ERROR TYPE
000702' 010137  021026                  MOV     R1,@#MDMAW0             ;LOAD PCBB+2 WITH FAILING ADDRESS
000706' 010337  021026                  MOV     R3,@#MDMAW0             ;LOAD PCBB+4 WITH GOOD DATA
000712' 010437  021026                  MOV     R4,@#MDMAW0             ;LOAD PCBB+6 WITH BAD DATA
000716' 000261                          SEC                             ;INDICATE THIS MICROTEST FAILED
000720' 000434                          BR      45$

000722' 005767  000164          30$:    TST     BOTH                    ;HAVE WE DONE BOTH HALVES YET?
000726' 001030                          BNE     40$                     ;YES
000730' 005267  000156                  INC     BOTH                    ;NO, WE ARE ABOUT TO DO OTHER HALF
000734' 012767  010000  000142          MOV     #WCSSIZ/2,LOWLIM        ;SET LOW LIMIT AT BOTTOM OF UPPER HALF
000742' 012767  020000  000136          MOV     #WCSSIZ,HILIM           ;HIGH LIMIT IS TOP OF UPPER HALF
```

```
000750' 010700                                  MOV     PC,R0                   ;GET BASE ADDRESS OF THIS MICROTEST
000752' 062700  177026                          ADD     #MICROB-.,R0
000756' 016701  001330                          MOV     MICBSZ,R1               ;GET ITS SIZE
000762' 012702  001000                          MOV     #1000,R2                ;RELOCATE TO JUST ABOVE STACK
000766' 112022                          35$:    MOVB    (R0)+,(R2)+             ;RELOCATE THIS MICROTEST
000770' 005301                                  DEC     R1
000772' 001375                                  BNE     35$
000774' 010700                                  MOV     PC,R0
000776' 062700  177610                          ADD     #5$-.,R0
001002' 162700  007000                          SUB     #WCSSIZ/2-1000,R0
001006' 000110                                  JMP     (R0)                    ;GO DO THE UPPER HALF TEST
001010' 000241                          40$:    CLC                             ;INDICATE THIS MICROTEST SUCCESS
001012' 106746                          45$:    MFPS    -(SP)                   ;SAVE RESULT STATUS
001014' 005767  000072                          TST     BOTH                    ;WE DONE BOTH HALVES?
001020' 001412                                  BEQ     55$                     ;NO
001022' 010700                                  MOV     PC,R0                   ;YES, WE HAVE TO RELOCATE BACK TO...
001024' 062700  176754                          ADD     #MICROB-.,R0            ;WHERE WE WERE
001030' 016701  001256                          MOV     MICBSZ,R1
001034' 012702  010000                          MOV     #WCSSIZ/2,R2
001040' 112022                          50$:    MOVB    (R0)+,(R2)+
001042' 005301                                  DEC     R1
001044' 001375                                  BNE     50$
001046' 106426                          55$:    MTPS    (SP)+                   ;RESTORE RESULT STATUS
001050' 112767  000001  177457                  MOVB    #1,PCSR1+1              ;TELL HOST WHICH TEST THIS WAS
001056' 016737  177452  021020                  MOV     PCSR1,@#IPCSR1
001064' 000207                                  RTS     PC                      ;RETURN TO MICROMONITOR

001066' 000026                          B1STBL: MICB1A-.
001070' 000162                                  MICB1B-.
001072' 000320                                  MICB1C-.
001074' 000432                                  MICB1D-.
001076' 000670                                  MICB1E-.
001100' 001002                                  MICB1F-.
001102' 000000                                  .WORD   0                       ;END OF LIST

001104' 000000                          LOWLIM: .WORD   0                       ;LOW LIMIT OF MEMORY TESTS
001106' 000000                          HILIM:  .WORD   0                       ;HIGH LIMIT OF MEMORY TESTS
001110'    000                          SUBNUM: .BYTE   0                       ;CURRENT SUBTEST # BEING EXECUTED
001111'    000                          ERRTYP: .BYTE   0                       ;TYPE OF ERROR, DATA =0, ADDRESS=1
001112' 000000                          BOTH:   .WORD   0                       ;FLAG INDICATING WHEN BOTH HALVES OF
                                                                                ;WCS HAVE BEEN TESTED
```

```
                                        .SBTTL  MODULE B, MICROTEST #1, MICROSUBTEST A

                                        ;*********************************************************************
                                        ;
                                        ;THIS IS AN ACCESS TEST OF WCS MEMORY. IT WRITES ONES
                                        ;IN MEMORY BETWEEN LOWLIM AND HILIM AND VERIFIES SAME. IT THEN WRITES
                                        ;ZEROS AND VERIFIES SAME. IT ALSO CHECKS FOR BOGUS PARITY ERRORS.
                                        ;
                                        ;*********************************************************************

001114' 010046                          MICB1A: MOV     R0,-(SP)                ;SAVE R0
001116' 112767  000001  177764                  MOVB    #1,SUBNUM               ;TELL WE ARE IN SUBTEST A
001124' 106427  000300                          MTPS    #PRI06                  ;ALLOW PARITY ERRORS
001130' 012703  177777                          MOV     #177777,R3              ;GOOD DATA = ALL ONES
001134' 016700  177744                  5$:     MOV     LOWLIM,R0               ;GET BASE ADDRESS OF MEMORY
001140' 010320                          10$:    MOV     R3,(R0)+                ;WRITE ALL OVER MEMORY
001142' 020067  177740                          CMP     R0,HILIM
001146' 103774                                  BLO     10$

001150' 016701  177730                          MOV     LOWLIM,R1               ;POINT BACK TO BASE
001154' 011104                          12$:    MOV     (R1),R4                 ;READ DATA
001156' 032767  000004  177344                  BIT     #PARFLG,FLG2            ;DID A PARITY ERROR OCCUR?
001164' 001407                                  BEQ     15$                     ;NO
001166' 020304                                  CMP     R3,R4                   ;YES, WAS DATA READ GOOD?
001170' 001007                                  BNE     20$                     ;NO, DATA ERROR
001172' 112767  010000  177711                  MOVB    #PARERR,ERRTYP          ;TELL MICROMONITOR TYPE OF ERROR
001200' 000261                                  SEC                             ;TELL MICROMONITOR ERROR OCCURRED
001202' 000421                                  BR      40$                     ;LEAVE

001204' 020304                          15$:    CMP     R3,R4                   ;WAS DATA GOOD?
001206' 001405                                  BEQ     30$                     ;YES
001210' 112767  000000  177673          20$:    MOVB    #DATERR,ERRTYP          ;TELL MICROMONITOR TYPE OF ERROR
001216' 000261                                  SEC                             ;TELL MICROMONITOR ERROR OCCURRED
001220' 000412                                  BR      40$

001222' 062701  000002                  30$:    ADD     #2,R1                   ;POINT TO NEXT MEMORY ADDRESS
001226' 020167  177654                          CMP     R1,HILIM                ;DONE ALL MEMORY?
001232' 001350                                  BNE     12$                     ;NOT YET
001234' 005703                                  TST     R3                      ;DONE BOTH DATA TYPES?
001236' 001402                                  BEQ     35$                     ;YES
001240' 005103                                  COM     R3                      ;NO, WRITE ONES NOW
001242' 000734                                  BR      5$

001244' 000241                          35$:    CLC                             ;INDICATE SUCCESS
001246' 012600                          40$:    MOV     (SP)+,R0                ;RESTORE R0
001250' 000207                                  RTS     PC
```

```
                                        .SBTTL  MODULE B, MICROTEST #1, MICROSUBTEST B

                                        ;*************************************************************************
                                        ;
                                        ;ADDRESS SHIFT TEST
                                        ;
                                        ;THIS TEST ASSUMES ALL MEMORY BETWEEN LOWLIM AND HILIM IS ZEROS
                                        ;IT CHECKS FOR PROPER BANK SELECTION BY WRITING 1'S IN A LOCATION
                                        ;AND CHECKING FOR 1'S IN THE SAME LOCATION OF OTHER 4K BANKS (ADDRESS ERROR).
                                        ;IT ALSO CHECKS FOR DATA ERRORS I.E. NON-ZERO DATA IN LOCATIONS NOT
                                        ;WRITTEN WITH 1'S AND MAKES SURE LOCATIONS WRITTEN WITH 1'S HAVE 1'S IN THEM
                                        ;
                                        ;*************************************************************************

001252' 010046                          MICB1B: MOV     R0,-(SP)                ;SAVE R0
001254' 112767  000002  177626                  MOVB    #2,SUBNUM               ;TELL HIM WE ARE IN THE SECOND SUBTEST

001262' 016700  177616                          MOV     LOWLIM,R0               ;SET 'WRITTEN TO' ADDRESS
001266' 016701  177612                  10$:    MOV     LOWLIM,R1               ;SET 'READ FROM' ADDRESS
001272' 012710  177777                          MOV     #177777,(R0)            ;WRITE DATA INTO LOCATION AT BASE

001276' 011104                          20$:    MOV     (R1),R4                 ;READ DATA FROM MEMORY WHICH IS AN EVEN
                                                                                ;INCREMENT AWAY
001300' 020001                                  CMP     R0,R1                   ;IS 'READ FROM AND 'WRITTEN TO'
                                                                                ;ADDRESSES THE SAME?
001302' 001413                                  BEQ     70$                     ;YES, SO DATA IN BOTH SHOULD BE THE SAME
001304' 005003                                  CLR     R3                      ;GOOD DATA IS ZEROS
001306' 005704                                  TST     R4                      ;DATA IN 'READ FROM' MUST BE 0'S
001310' 001422                                  BEQ     50$                     ;OK GO CHANGE 'READ FROM' ADDRESS
                                                                                ;ERROR OCCURRED BUT WE DON'T KNOW IF...
                                                                                ;IT IS A DATA ERROR OR AN ADDRESS ERROR
001312' 022704  177777                          CMP     #177777,R4              ;WAS DATA READ ALL 1'S?
001316' 001012                                  BNE     40$                     ;NO, SO IT WAS A DATA ERROR
001320' 112767  000001  177563                  MOVB    #ADRERR,ERRTYP          ;YES, IT WAS AN ADDRESS ERROR
001326' 000261                                  SEC                             ;INDICATE FAILURE
001330' 000426                                  BR      60$                     ;LEAVE THIS SUBTEST
001332' 022704  177777                  70$:    CMP     #177777,R4              ;DATA READ MUST BE ALL 1'S
001336' 001407                                  BEQ     50$                     ;IT IS GOOD
001340' 012703  177777                          MOV     #177777,R3              ;GOOD DATA IS ONES
001344' 112767  000000  177537          40$:    MOVB    #DATERR,ERRTYP          ;IT WAS A DATA ERROR
001352' 000261                                  SEC                             ;INDICATE FAILURE
001354' 000414                                  BR      60$                     ;LEAVE THIS SUBTEST

001356' 062701  001000                  50$:    ADD     #1000,R1                ;CHANGE 'READ FROM' ADDRESS BY .25K
001362' 020167  177520                          CMP     R1,HILIM                ;WE GO PAST BOUNDARY?
001366' 103743                                  BLO     20$                     ;NO CONTINUE WITH SAME 'WRITTEN TO'
001370' 005010                                  CLR     (R0)                    ;CLEAR OLD 'WRITTEN TO' ADDRESS
001372' 062700  001000                          ADD     #1000,R0                ;CHANGE 'WRITTEN TO' ADDRESS BY .25K
001376' 020067  177504                          CMP     R0,HILIM                ;WE PAST BOUNDARY?
001402' 103731                                  BLO     10$                     ;NO CONTINUE THIS SUBTEST
001404' 000241                                  CLC                             ;INDICATE SUCCESS OF THIS SUBTEST
001406' 012600                          60$:    MOV     (SP)+,R0                ;RESTORE R0
001410' 000207                                  RTS     PC
```

```
                                     .SBTTL  MODULE B, MICROTEST #1, MICROSUBTEST C

                                     ;*************************************************************************
                                     ;
                                     ;DATA LATCH TEST
                                     ;
                                     ;AT THE FIRST LOCATION OF EACH BANK A '1' IS SHIFTED THROUGH EACH
                                     ;BIT POSITION AND CHECKED FROM LSB TO MSB.
                                     ;THEN IN THE SAME LOCATION A '0' IS SHIFTED THROUGH EACH BIT POSITION
                                     ;AND CHECKED.
                                     ;
                                     ;*************************************************************************


001412' 010046                       MICB1C: MOV     R0,-(SP)                ;SAVE R0
001414' 112767  000003  177466               MOVB    #3,SUBNUM               ;TELL WEA ARE IN SUBTEST 'C'
001422' 016701  177456                       MOV     LOWLIM,R1               ;GET BASE ADDRESS OF MEMORY
001426' 012700  000001               1$:     MOV     #1,R0                   ;DATA = 1 IN LEAST SIGNIFICANT BIT
001432' 010002                               MOV     R0,R2                   ;INDICATE WE ARE SHIFTING A '1'
001434' 010011                       2$:     MOV     R0,(R1)                 ;WRITE LOCATION WITH GOOD DATA
001436' 011104                               MOV     (R1),R4                 ;READ DATA FROM SAME LOCATION
001440' 020004                       3$:     CMP     R0,R4                   ;IS DATA THE SAME AS WRITTEN?
001442' 001406                               BEQ     4$                      ;YES, OK
001444' 112767  000000  177437               MOVB    #DATERR,ERRTYP          ;ERROR IS DATA ERROR
001452' 010003                               MOV     R0,R3                   ;GOOD DATA
001454' 000261                               SEC                             ;INDICATE THIS SUBTEST FAILED
001456' 000421                               BR      6$                      ;LEAVE THIS SUBTEST
001460' 005702                       4$:     TST     R2                      ;ARE WE SHIFTING A 1 OR A 0?
001462' 001406                               BEQ     5$                      ;ZERO
001464' 006300                               ASL     R0                      ;SHIFT THE ONE OVER
001466' 103362                               BCC     2$                      ;IF THE '1' HAS NOT BEEN SHIFTED...
                                                                             ;THRU THE 16 POSITIONS CONTINUE WITH 1
001470' 005002                               CLR     R2                      ;ELSE START SHIFTING A '0'
001472' 012700  177776                       MOV     #177776,R0              ;START WITH LSB = 0 ALL OTHERS 1'S
001476' 000756                               BR      2$                      ;CONTINUE WITH SHIFTING A 0
001500' 000261                       5$:     SEC                             ;MOVE '0' OVER ONE BIT POSITION
001502' 006100                               ROL     R0                      ;HAS '0' BEEN IN ALL POSITIONS?
001504' 103753                               BCS     2$                      ;NO CONTINUE WITH SHIFTING A 0
001506' 062701  001000                       ADD     #1000,R1                ;CONTINUE TEST AT NEXT BOUNDARY
001512' 020167  177370                       CMP     R1,HILIM                ;DONE ALL OF MEMORY?
001516' 103743                               BLO     1$                      ;NO
001520' 000241                               CLC                             ;INDICATE THIS SUBTEST SUCCESSFUL
001522' 012600                       6$:     MOV     (SP)+,R0                ;RESTORE R0
001524' 000207                               RTS     PC
```

```
                                        .SBTTL  MODULE B, MICROTEST #1, MICROSUBTEST D

                                        ;*********************************************************************
                                        ;
                                        ;ADDRESS BIT SHIFT #1
                                        ;
                                        ;THIS TEST CHECKS FOR DUAL ADDRESS PROBLEMS BY FIRST WRITING A BACKGROUND
                                        ;PATTERN THROUGHOUT MEMORY.
                                        ;THEN STARTING AT THE LOWEST LOCATION IN A BANK IT WRITES THE COMPLEMENT
                                        ;OF THE BACKGROUND PATTERN. THEN READS THE MEMORY FOR CORRECT CONTENTS
                                        ;IT THEN SHIFTS A '1' THROUGH THE ADDRESS POINTER AND REPEATS THE ABOVE.
                                        ;IT DOES THIS FOR EACH BANK.
                                        ;
                                        ;*********************************************************************

001526' 010046                          MICB1D: MOV     R0,-(SP)                ;SAVE R0
001530' 112767  000004  177352                  MOVB    #4,SUBNUM               ;TELL WE ARE IN SUBTEST 4
001536' 012767  000377  177004                  MOV     #377,BAKPAT             ;LOAD BAKPAT CONSTANT
001544' 005003                          1$:     CLR     R3                      ;CONTAINS BANK ADDRESS
001546' 016700  177332                  2$:     MOV     LOWLIM,R0               ;WRITE LINK MEMORY WITH BACKGROUND...
001552' 016720  176772                  3$:     MOV     BAKPAT,(R0)+            ;PATTERN
001556' 020067  177324                          CMP     R0,HILIM
001562' 103773                                  BLO     3$
001564' 016700  176760                          MOV     BAKPAT,R0               ;R0 CONTAINS GOOD DATA

001570' 005002                          4$:     CLR     R2                      ;R2 WILL BE OUR 'WRITTEN TO' ADDRESS
001572' 050302                          6$:     BIS     R3,R2                   ;INDEX INTO THIS BANK
001574' 020267  177304                          CMP     R2,LOWLIM               ;IS RESULT LESS THAN MEM BASE?
001600' 103450                                  BLO     16$                     ;YES, DON'T USE THIS ADDRESS
001602' 020267  177300                          CMP     R2,HILIM                ;IS RESULT LARGER THAN MEM TOP?
001606' 103061                                  BHIS    20$                     ;YES, DON'T USE THIS ADDRESS EITHER
001610' 000312                                  SWAB    (R2)                    ;WRITE COMPLEMENT OF PATTERN
001612' 005001                                  CLR     R1                      ;R1 WILL BE OUR 'READ FROM' ADDRESS
001614' 050301                          7$:     BIS     R3,R1                   ;INDEX INTO THIS BANK
001616' 020167  177262                          CMP     R1,LOWLIM               ;IS RESULT LESS THAN  MEM BASE?
001622' 103426                                  BLO     12$                     ;YES, DON'T USE THIS ADDRESS
001624' 020167  177256                          CMP     R1,HILIM                ;IS RESULT LARGER THAN MEM TOP?
001630' 103033                                  BHIS    15$                     ;YES, DON'T USE THIS ADDRESS EITHER
001632' 011104                                  MOV     (R1),R4                 ;READ DATA
001634' 020102                                  CMP     R1,R2                   ;IS 'READ FROM' AND 'WRITTEN TO' SAME?
001636' 001414                                  BEQ     10$                     ;YES, GO CHECK DATA
001640' 020004                                  CMP     R0,R4                   ;NO, DATA READ SHOULD BE SAME AS BAKPAT
001642' 001416                                  BEQ     12$                     ;IF SO CONTINUE WITH NEW INDEX

001644' 112767  000001  177237  8$:     MOVB    #ADRERR,ERRTYP          ;INDICATE ADDRESS ERROR
001652' 010003                                  MOV     R0,R3                   ;GET GOOD DATA
001654' 000261                                  SEC                             ;INDICATE THIS SUBTEST FAILED
001656' 000441                                  BR      25$                     ;LEAVE THIS SUBTEST
001660' 020067  176664                          CMP     R0,BAKPAT               ;DOES R0 CONTAIN SWAPPED DATA?
001664' 001405                                  BEQ     12$                     ;NO
001666' 000403                                  BR      11$                     ;YES

001670' 000300                          10$:    SWAB    R0                      ;MAKE GOOD DATA LIKE SWAPPED BAKPAT
001672' 020004                                  CMP     R0,R4                   ;IS DATA READ SAME AS DATA WRITTEN?
001674' 001363                                  BNE     8$                      ;NO, ERROR
001676' 000300                          11$:    SWAB    R0                      ;RESTORE GOOD DATA TO BAKPAT
```

```
001700' 040301          12$:    BIC     R3,R1           ;REMOVE BANK SELECT FROM 'READ FROM'
001702' 005701                  TST     R1              ;DID AN INDEX EXIST?
001704' 001001                  BNE     13$             ;YES
001706' 005201                  INC     R1              ;NO, MAKE ONE
001710' 006101          13$:    ROL     R1              ;SHIFT INDEX OVER
001712' 020127  001000          CMP     R1,#1000        ;IS IT LESS THAN .25K
001716' 103736                  BLO     7$              ;YES, CONTINUE CHECKING FOR BOGUS DATA
                                                        ;IN THIS BANK
001720' 000312          15$:    SWAB    (R2)            ;RESTORE CONTENTS OF 'WRITTEN TO'
001722' 040302          16$:    BIC     R3,R2           ;REMOVE BANK SELECT FROM 'WRITTEN TO'
001724' 005702                  TST     R2              ;DOES 'WRITTEN TO' CONTAIN AN ADDRESS?
001726' 001001                  BNE     18$             ;YES
001730' 005202                  INC     R2              ;NO, MAKE ONE
001732' 006102          18$:    ROL     R2              ;SHIFT 'WRITTEN TO' ADDRESS
001734' 020227  001000          CMP     R2,#1000        ;IS 'WRITTEN TO' LARGER THAN .25K?
001740' 103714                  BLO     6$              ;NO
001742' 060203                  ADD     R2,R3           ;MAKE 'WRITTEN TO' POINT TO NEXT .25K
001744' 020367  177136          CMP     R3,HILIM        ;IS RESULT LARGER THAN MEM TOP?
001750' 103707                  BLO     4$              ;NO CONTINUE TESTING
001752' 000367  176572  20$:    SWAB    BAKPAT          ;REPEAT TEST WITH COMPLEMENTED PATTERN
001756' 001672                  BEQ     1$
001760' 000241                  CLC                     ;INDICATE THIS SUBTEST SUCCESS
001762' 012600          25$:    MOV     (SP)+,R0        ;RESTORE R0
001764' 000207                  RTS     PC
```

```
                                  .SBTTL  MODULE B, MICROTEST #1, MICROSUBTEST E

                                  ;*****************************************************************************
                                  ;
                                  ;ADDRESS BIT SHIFT #2
                                  ;
                                  ;THIS TEST CHECKS FOR DUAL ADDRESSING PROBLEMS BY WRITING EACH LOCATION
                                  ;OF MEMORY WITH ITS ADDRESS AND THEN VERIFYING EACH LOCATION.
                                  ;IT THEN DOES THE SAME THING BUT USES THE COMPLEMENT OF THE ADDRESS
                                  ;AS DATA.
                                  ;
                                  ;**************************************************************************

001766' 010046                    MICB1E: MOV     R0,-(SP)                ;SAVE R0
001770' 112767  000005  177112            MOVB    #5,SUBNUM               ;TELL WHICH SUBTEST WE ARE IN
001776' 005003                            CLR     R3                      ;FLAG INDICATING ADDRESS IS COMPLEMENTED
002000' 016701  177100                    MOV     LOWLIM,R1               ;GET STARTING ADDRESS OF MEMORY
002004' 010100                    1$:     MOV     R1,R0                   ;GET ADDRESS TO WORK WITH
002006' 005703                            TST     R3                      ;SHOULD WE COMPLEMENT THE DATA TO STORE?
002010' 001401                            BEQ     2$                      ;NO STORE AS IS
002012' 005100                            COM     R0                      ;COMPLEMENT DATA
002014' 010021                    2$:     MOV     R0,(R1)+                ;WRITE DATA
002016' 020167  177064                    CMP     R1,HILIM                ;IS NEW ADDRESS LARGER THAN MEM?
002022' 103770                            BLO     1$                      ;NO, KEEP FILLING MEMORY

002024' 014104                    3$:     MOV     -(R1),R4                ;READ DATA STORED
002026' 020004                            CMP     R0,R4                   ;IS DATA READ SAME AS WRITTEN?
002030' 001406                            BEQ     4$                      ;YES
002032' 112767  000001  177051            MOVB    #ADRERR,ERRTYP          ;INDICATE ADDRESS ERROR
002040' 010003                            MOV     R0,R3                   ;GET GOOD DATA
002042' 000261                            SEC                             ;INDICATE FAILURE
002044' 000414                            BR      10$                     ;LEAVE THIS SUBTEST
002046' 010100                    4$:     MOV     R1,R0                   ;CALC GOOD DATA FOR NEXT LOCATION
002050' 162700  000002                    SUB     #2,R0
002054' 005703                            TST     R3
002056' 001401                            BEQ     5$
002060' 005100                            COM     R0
002062' 020167  177016            5$:     CMP     R1,LOWLIM               ;HAVE WE CHECKED ALL LOCATIONS
002066' 101356                            BHI     3$                      ;NOT YET
002070' 005103                            COM     R3                      ;HAVE WE DONE IT COMPLEMENTED?
002072' 001344                            BNE     1$                      ;NO, REPEAT WITH COMPLEMENT
002074' 000241                            CLC                             ;INDICATE SUCCESS
002076' 012600                    10$:    MOV     (SP)+,R0                ;RESTORE R0
002100' 000207                            RTS     PC
```

```
                                        .SBTTL  MODULE B, MICROTEST #1, MICROSUBTEST F

                                        ;****************************************************************************
                                        ;
                                        ;MARCH TEST
                                        ;
                                        ;THIS TEST WRITES A BACKGROUND PATTERN IN ALL OF MEMORY
                                        ;
                                        ;1-READ EVERY LOCATION FOR CORRECT DATA, SWAPS BYTES AT EACH LOCATION
                                        ;AND PROCEED IN MAX TO MIN DIRECTION
                                        ;2-READ EVERY LOCATION FOR SWAPPED DATA, WRITES ORIGINAL PATTERN IN EACH
                                        ;LOCATION AND PROCEEDS IN MIN TO MAX DIRECTION
                                        ;3-REPEAT 1 GOING IN MIN TO MAX DIRECTION
                                        ;4-REPEAT 2 GOING IN MAX TO MIN DIRECTION
                                        ;
                                        ;****************************************************************************

002102' 010046                          MICB1F: MOV     R0,-(SP)                ;SAVE R0
002104' 112767  000006  176776                  MOVB    #6,SUBNUM               ;TELL WHICH SUBTEST WE ARE IN
002112' 005003                                  CLR     R3                      ;ADDRESS DIRECTION FLAG 0 = MIN.->MAX
002114' 016701  176764                  10$:    MOV     LOWLIM,R1               ;FILL  MEMORY WITH BACKGROUND PATTERN
002120' 012700  000377                          MOV     #377,R0                 ;BACKGROUND PATTERN=LOW BYTE ALL 1'S
002124' 010021                          12$:    MOV     R0,(R1)+
002126' 020167  176754                          CMP     R1,HILIM
002132' 103774                                  BLO     12$

002134' 014104                          20$:    MOV     -(R1),R4                ;STARTING FROM THE TOP, READ DATA
002136' 020004                                  CMP     R0,R4                   ;R0 = GOOD DATA, R4 = DATA READ
002140' 001406                                  BEQ     30$                     ;IF SAME OK
002142' 112767  000000  176741                  MOVB    #DATERR,ERRTYP          ;INDICATE DATA ERROR
002150' 010003                                  MOV     R0,R3                   ;GET GOOD DATA
002152' 000261                                  SEC                             ;INDICATE FAILURE
002154' 000454                                  BR      200$                    ;LEAVE THIS SUBTEST
002156' 000300                          30$:    SWAB    R0                      ;NEW GOOD DATA PATTERN
002160' 010011                                  MOV     R0,(R1)                 ;STORE AT SAME PLACE
002162' 011104                                  MOV     (R1),R4                 ;READ IT BACK
002164' 020400                                  CMP     R4,R0                   ;R0=GOOD DATA, R4=DATA READ
002166' 001406                                  BEQ     40$                     ;IF SAME OK
002170' 112767  000000  176713                  MOVB    #DATERR,ERRTYP          ;INDICATE DATA ERROR
002176' 010003                                  MOV     R0,R3                   ;GET GOOD DATA
002200' 000261                                  SEC                             ;FAILURE
002202' 000441                                  BR      200$                    ;LEAVE THIS SUBTEST
002204' 000300                          40$:    SWAB    R0                      ;SWITCH GOOD DATA AGAIN
002206' 001027                                  BNE     90$                     ;IF ORIGINAL PATTERN THEN WE ARE...
                                                                                ;READING THE MEMORY TO CONTAIN A...
                                                                                ;BACKGROUND OF LOW BYTE = ALL 1'S...
                                                                                ;AND WRITING IT BACK SWAPPED AND...
                                                                                ;VERIFING SWAPPED DATA
002210' 005703                          50$:    TST     R3                      ;
002212' 001027                                  BNE     100$
002214' 062701  000002                  60$:    ADD     #2,R1                   ;WE ARE GOING MIN->MAX SO ADJUST POINTER
002220' 020167  176662                          CMP     R1,HILIM                ;WE AT MAX?
002224' 103012                                  BHIS    80$                     ;YES
002226' 011104                          70$:    MOV     (R1),R4                 ;READ DATA
002230' 020004                                  CMP     R0,R4                   ;R0=GOOD DATA, R4=DATA READ
002232' 001751                                  BEQ     30$                     ;OK IF SAME
```

```
002234' 112767  000000  176647            MOVB    #DATERR,ERRTYP          ;INDICATE DATA ERROR
002242' 010003                            MOV     R0,R3                   ;GET GOOD DATA
002244' 000261                            SEC                             ;INDICATE FAILURE
002246' 000417                            BR      200$                    ;LEAVE THIS SUBTEST
002250' 000742                            BR      30$
002252' 000300                    80$:    SWAB    R0                      ;SWITCH GOOD DATA
002254' 001727                            BEQ     20$                     ;IF LOW BYTE = ALL 0'S GO IN MAX->MIN
002256' 005103                            COM     R3                      ;ELSE GO IN MIN->MAX DIRECTION
002260' 016701  176620                    MOV     LOWLIM,R1               ;SET ADDRESS POINTER TO MIN
002264' 000760                            BR      70$

002266' 005703                    90$:    TST     R3                      ;ARE WE GOING MIN->MAX?
002270' 001347                            BNE     50$                     ;YES

002272' 020167  176606            100$:   CMP     R1,LOWLIM               ;NO, CHECK TO SEE IF WE ARE AT MIN
002276' 101316                            BHI     20$                     ;NOT YET
002300' 000300                            SWAB    R0                      ;WE ARE AT MIN SO SWITCH GOOD DATA
002302' 001751                            BEQ     70$                     ;IF LOW BYTE = ALL 0'S ELSE WE ARE DONE
002304' 000241                            CLC                             ;INDICATE SUCCESS
002306' 012600                    200$:   MOV     (SP)+,R0                ;RESTORE R0
002310' 000207                            RTS     PC

002312' 002314                    MICBSZ::MICBSZ-MICROB+2                 ;SIZE OF MICROCODE MODULE B
        000001                    .END
```

```
ADDREG= 177774          103#
ADRERR= 000001          101#   414    524    589
BAKPAT  000550R   002   235#   497*   500    503    528    555*
BIT0  = 000001           51#    81
BIT1  = 000002           50#    82
BIT10 = 002000           41#    53     70
BIT11 = 004000           40#    69
BIT12 = 010000           39#    53     68     89
BIT13 = 020000           38#    67     88
BIT14 = 040000           37#    66     87    210
BIT15 = 100000           36#    65     86    215    218
BIT2  = 000004           49#    83
BIT3  = 000010           48#    84
BIT4  = 000020           47#    85
BIT5  = 000040           46#
BIT6  = 000100           45#
BIT7  = 000200           44#
BIT8  = 000400           43#    53     71
BIT9  = 001000           42#
BOTH    001112R   002   258*   288    290*   306    332#
B1STBL  001066R   002   263    320#
CLRFIF= 021034           22#
CMDREG= 177776          104#   119*
CSRFLG= 000001           81#   167    198
CSRVEC= 000064           74#   144*   145*
CSRWRT  000376R   002   143    198#
DATERR= 000000          100#   365    420    459    635    644    662
DMACSR= 021002            9#   209    218*
DMAF  = 021022           17#
DMAINT  000432R   002   138    209#
DMAT0 = 021004           10#
DMAT1 = 021006           11#
DMAVEC= 000114           75#   139*   140*
DMAWC = 021024           18#
DMDONE  000546R   002   209*   210    215    234#
DNI   = 004000           69#   123    191
ERRFLG= 000002           82#   201
ERRINT  000406R   002   125    201#
ERRTYP  001111R   002   331#   359*   365*   414*   420*   459*   524*   589*   635*   644*   662*
FATI  = 000400           71#
FLG2    000530R   002   160*   163    167    192*   198*   201*   222*   229#   265*   355
HILIM   001106R   002   260*   292*   329#   350    370    425    429    475    501    509    516    553    583
                        629    657
INERR = 000003           80#   189
INMON = 000001           78#   121    175    187
INTST = 000002           79#   256
IOADR = 020000           97#    98
IOSIZ = 020000           93#    98
IPCSR0= 021000            8#   123*   172    191*   202*   212*   217*   223*
IPCSR1= 021020           16#   122*   175*   190*   257*   317*
IPCSR2  000536R   002   158*   159*   232#   274    275
LASFTP= 012400           53#
LFRBUF= 021032           21#
LINADR= 100000           99#   102
LINSIZ= 077774           95#   102
LOWLIM  001104R   002   259*   291*   328#   348    353    398    399    452    499    507    514    577    598
```

```
                              626     670     676
LRBUF  = 021044                26#
LTAC   = 021030                20#
MDMAR0= 021014                 14#    276     277
MDMAR1= 021016                 15#
MDMAW0= 021026                 19#    280*    282*    283*    284*
MDMAW1= 021036                 23#
MDMA0  = 021010                12#    274*    278*
MDMA1  = 021012                13#    275*    279*
MDMSR  = 021042                25#
MICBSZ  002312RG     002      295     310     684#
MICB1   000552R      002      227     256#
MICB1A  001114R      002      320     344#
MICB1B  001252R      002      321     395#
MICB1C  001412R      002      322     450#
MICB1D  001526R      002      323     495#
MICB1E  001766R      002      324     574#
MICB1F  002102R      002      325     623#
MICROB  000000RG     002      118#    294     309     684
MODREG= 177774                102#    103     104
NPRERR= 100000                 86#    217
NPRFLG= 000020                 85#
NXMERR= 040000                 87#    212
NXMFLG= 000010                 84#
PARERR= 010000                 89#    223     359
PARFLG= 000004                 83#    222     355
PARINT  000510R      002      133     222#
PARVEC= 000140                 76#    134*    135*
PCBADR  000542R      002      233#
PCEI   = 040000                66#
PCSRSW= 021040                 24#    152
PCSR1   000534R      002      121*    122     187*    189*    190     231#    256*    257     316*    317
PHYAD0= 021060                 27#
PHYAD1= 021062                 28#
PHYAD2= 021064                 29#
PHYAD3= 021066                 30#
PHYAD4= 021070                 31#
PHYAD5= 021072                 32#
PRI00  = 000000                61#    161*
PRI01  = 000040                60#
PRI02  = 000100                59#
PRI03  = 000140                58#
PRI04  = 000200                57#    145
PRI05  = 000240                56#    150
PRI06  = 000300                55#    346*
PRI07  = 000340                54#    118*    128     135     140     166*
RCEI   = 002000                70#
ROMADR= 040000                 98#     99
ROMSIZ= 040000                 94#     99
RXI    = 020000                67#
SANTIM  000532R      002      206*    230#
SANVEC= 000134                 73#    149*    150*
SERI   = 100000                65#
SIZ4K  = 020000                90#     91      92      93
SIZ8K  = 040000                91#     94      95
STACK  = 001000                77#    120
```

```
SUBNUM  001110R   002   280   330#  345*  396*  451*  496*  575*  624*
TBLB    000526R   002   181   227#
TIMINT  000424R   002   148   206#
TXI   = 010000           68#
UNIERR= 020000           88#  202
WCSADR= 000000           96#   97
WCSSIZ= 020000           92#   97   260   291   292   302   311
.     = 002314R   002   125   133   138   143   148   169   181   204   219   225   227   263   294
                        301   309   320   321   322   323   324   325
```

```
BCOMPL     1#
BERROR     1#
BGNAU      1#
BGNAUT     1#
BGNCLN     1#
BGNDU      1#
BGNHRD     1#
BGNHW      1#
BGNINI     1#
BGNMOD     1#
BGNMSG     1#
BGNPRO     1#
BGNPTA     1#
BGNRPT     1#
BGNSEG     1#
BGNSET     1#
BGNSFT     1#
BGNSRV     1#
BGNSUB     1#
BGNSW      1#
BGNTST     1#
BNCOMP     1#
BNERRO     1#
BREAK      1#
BRESET     1#
CKLOOP     1#
CLOCK      1#
CLOSE      1#
CLRVEC     1#
COMMEN     1#
DELAY      1#
DESCRI     1#
DEVTYP     1#
DISPAT     1#
DISPLA     1#
DOCLN      1#
DODU       1#
DORPT      1#
ENDAU      1#
ENDAUT     1#
ENDCLN     1#
ENDCOM     1#
ENDDU      1#
ENDHRD     1#
ENDHW      1#
ENDINI     1#
ENDMOD     1#
ENDMSG     1#
ENDPRO     1#
ENDPTA     1#
ENDRPT     1#
ENDSEG     1#
ENDSET     1#
ENDSFT     1#
ENDSRV     1#
ENDSUB     1#
```

```
ENDSW       1#
ENDTST      1#
EQUALS      1#
ERRDF       1#
ERRHRD      1#
ERROR       1#
ERRSF       1#
ERRSOF      1#
ERRTBL      1#
ESCAPE      1#
EXIT        1#
FEQUAL      1#
GETBYT      1#
GETPRI      1#
GETWOR      1#
GMANIA      1#
GMANID      1#
GMANIL      1#
GPHARD      1#
GPRMA       1#
GPRMD       1#
GPRML       1#
HEADER      1#
INLOOP      1#
IOSETU      1#
IOSTAR      1#
KT11        1#
LASTAD      1#
MANUAL      1#
MEMORY      1#
MSBYTE      1#
MSCHEC      1#
MSCNTO      1#
MSCOUN      1#
MSDATA      1#
MSDECR      1#
MSDEFA      1#
MSENDE      1#
MSERRI      1#
MSESCA      1#
MSESCS      1#
MSEXCP      1#
MSEXIT      1#
MSEXSE      1#
MSEXTJ      1#
MSGEN       1#
MSGENB      1#
MSGETS      1#
MSGETT      1#
MSGNGB      1#
MSGNIN      1#
MSGNLS      1#
MSGNSU      1#
MSGNTA      1#
MSGNTE      1#
MSNAPT      1#
```

```
MSHNAP      1#
MSINCR      1#
MSIOSE      1#
MSLDRO      1#
MSMASK      1#
MSMCHI      1#
MSMCLO      1#
MSMSK1      1#
MSPOP       1#
MSPRIN      1#
MSPUSH      1#
MSPUT       1#
MSPUT1      1#
MSRADI      1#
MSRBRO      1#
MSRMRO      1#
MSSETS      1#
MSSTAR      1#
MSSVC       1#
MSTLAB      1#
MSTSTL      1#
MSWORD      1#
MSXFER      1#
OPEN        1#
POINTE      1#
PRINTB      1#
PRINTF      1#
PRINTS      1#
PRINTX      1#
READBU      1#
READEF      1#
RFLAGS      1#
SETPRI      1#
SETVEC      1#
SLASH       1#
STARS       1#
SVC         1#
XFER        1#
XFERF       1#
XFERT       1#

. ABS.  000000     000
        000000     001
MICRB   002314     002

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROB.OBJ,MICROB.LST/CR/SOL/NL:TOC=SVC34R.P11,MICROB.MAC
 RUN-TIME: 2 3 .3 SECONDS
 RUN-TIME RATIO: 36/6=6.0
 CORE USED:  31K  (61 PAGES)
```

```

                        .TITLE  MICROC - MICROCODE MODULE C

        000000'         .CSECT MICRC

                        .SBTTL  REGISTER DEFINITIONS USED BY THE T11

        021000          IPCSR0  =       21000   ;INTERNAL PCSR0 ADDRESS
        021002          DMACSR  =       21002   ;DMA ENGINE CONTROL STATUS REGISTER
        021004          DMAT0   =       21004   ;DMA ENGINE TO ADDRESS REGISTER #0
        021006          DMAT1   =       21006   ;DMA ENGINE TO ADDRESS REGISTER #1
        021010          MDMA0   =       21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
        021012          MDMA1   =       21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
        021014          MDMAR0  =       21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
        021016          MDMAR1  =       21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
        021020          IPCSR1  =       21020   ;INTERNAL PCSR1 ADDRESS
        021022          DMAF    =       21022   ;DMA ENGINE FROM ADDRESS REGISTER
        021024          DMAWC   =       21024   ;DMA ENGINE WORD COUNT REGISTER
        021026          MDMAW0  =       21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
        021030          LTAC    =       21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
        021032          LFRBUF  =       21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
        021034          CLRFIF  =       21034   ;CLEAR FIFO
        021036          MDMAW1  =       21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
        021040          PCSRSW  =       21040   ;SWITCH PACK REGISTER
        021042          MDMSR   =       21042   ;MICROCPU DMA STATUS REGISTER
        021044          LRBUF   =       21044   ;LINK RECIEVE BUFFER COMPLETED
        021060          PHYAD0  =       21060   ;PHYSICAL ADDRESS ROM BYTE 0
        021062          PHYAD1  =       21062   ;PHYSICAL ADDRESS ROM BYTE 1
        021064          PHYAD2  =       21064   ;PHYSCIAL ADDRESS ROM BYTE 2
        021066          PHYAD3  =       21066   ;PHYSICAL ADDRESS ROM BYTE 3
        021070          PHYAD4  =       21070   ;PHYSICAL ADDRESS ROM BYTE 4
        021072          PHYAD5  =       21072   ;PHYSICAL ADDRESS ROM BYTE 5

                        .SBTTL  OTHER DEFINITIONS USED BY THE MICROCODE

        100000          BIT15   =       100000
        040000          BIT14   =       40000
        020000          BIT13   =       20000
        010000          BIT12   =       10000
        004000          BIT11   =       4000
        002000          BIT10   =       2000
        001000          BIT9    =       1000
        000400          BIT8    =       400
        000200          BIT7    =       200
        000100          BIT6    =       100
        000040          BIT5    =       40
        000020          BIT4    =       20
        000010          BIT3    =       10
        000004          BIT2    =       4
        000002          BIT1    =       2
        000001          BIT0    =       1
                        ;
        012400          LASFTP  =       BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
        000340          PRI07   =       340
        000300          PRI06   =       300
        000240          PRI05   =       240
```

```
000200          PRI04   =       200
000140          PRI03   =       140
000100          PRI02   =       100
000040          PRI01   =       40
000000          PRI00   =       0
                ;
                ;PCSR0 - PORT CONTROL STATUS REGISTER 0
                ;
100000          SERI    =       BIT15
040000          PCEI    =       BIT14
020000          RXI     =       BIT13
010000          TXI     =       BIT12
004000          DNI     =       BIT11
002000          RCEI    =       BIT10
000400          FATI    =       BIT8
                ;
000134          SANVEC= 134                 ;VECTOR ADDRESS FOR THE SANITY TIMER
000064          CSRVEC= 64                  ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
000114          DMAVEC= 114                 ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
000140          PARVEC= 140                 ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
001000          STACK=  1000                ;STACK LOCATION
000001          INMON=  1                   ;IN MICROMONITOR STATE
000002          INTST=  2                   ;IN A TEST STATE
000003          INERR=  3                   ;IN ERROR STATE
000001          CSRFLG= BIT0                ;CSR WRITE INTERRUPT OCCURED
000002          ERRFLG= BIT1                ;UNEXPECTED ERROR OCCURED
000004          PARFLG= BIT2                ;PARITY ERROR OCCURED
000010          NXMFLG= BIT3                ;NON-EXISTANT MEMORY ERROR OCCURRED
000020          NPRFLG= BIT4                ;NPR TIMEOUT ERROR OCCURRED
100000          NPRERR= BIT15               ;PCSR0 FLAG INDICATING NPR ERROR OCCURRED
040000          NXMERR= BIT14               ;PCSR0 FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
020000          UNIERR= BIT13               ;PCSR0 FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
010000          PARERR= BIT12               ;PCSR0 FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
020000          SIZ4K=  20000               ;4K WORDS
040000          SIZ8K=  SIZ4K*2             ;8K WORDS
020000          WCSSIZ= SIZ4K               ;SIZE OF WRITEABLE CONTROL STORE
020000          IOSIZ=  SIZ4K               ;SIZE OF I/O PAGE
040000          ROMSIZ= SIZ8K               ;SIZE OF ROM
077774          LINSIZ= SIZ8K*2-4                   ;SIZE OF LINK MEMORY
000000          WCSADR= 0                   ;BASE ADDRESS OF WCS
020000          IOADR=  WCSADR+WCSSIZ       ;BASE ADDRESS OF I/O PAGE
040000          ROMADR= IOADR+IOSIZ         ;BASE ADDRESS OF ROM
100000          LINADR= ROMADR+ROMSIZ       ;BASE ADDRESS OF LINK MEMORY
000000          DATERR= 0                   ;FLAG INDICATING DATA ERROR OCCURRED
000001          ADRERR= 1                   ;FLAG INDICATING ADDRESS ERROR OCCURRED
177774          MODREG= LINADR+LINSIZ       ;LINK MODE REGISTER
177774          ADDREG= MODREG              ;LINK STATION ADRESS RAM REGISTER
177776          CMDREG= MODREG+2            ;LINK COMMAND REGISTER
```

```

                                          .SBTTL    C_MODULE MICROCODE

                                          ;*********************************************************************
                                          ;
                                          ;THIS MODULE CONTAINS A MICROMONITOR AND MICROCODE FOR:
                                          ;
                                          ;         1-CLEAR INSTRUCTION
                                          ;         2-PCSR0 INTERRUPT BIT TEST
                                          ;         3-SANITY TIMER TEST
                                          ;         4-COMPREHENSIVE LINK MEMORY TEST
                                          ;         5-DMA 'TO' ADDRESS REGISTER TEST
                                          ;         6-DMA 'FROM' ADDRESS REGISTER TEST
                                          ;         7-DMA BLOCK TRANSFER TEST
                                          ;
                                          ;*********************************************************************


000000' 106427  000340                    MICROC::MTPS  #PRI07              ;DISABLE INTERRUPTS
000004' 012737  000000  177776                    MOV   #0,@#CMDREG         ;TURN OFF THE LINK
000012' 012706  001000                            MOV   #STACK,SP           ;SETUP STACK
000016' 012767  000001  000502                    MOV   #INMON,PCSR1        ;TELL HOST WE ARE IN MICROMONITOR
000024' 016737  000476  021020                    MOV   PCSR1,@#IPCSR1
000032' 010700                                    MOV   PC,R0               ;GET ADDRESS OF UNEXPECTED ERROR...
000034' 062700  000326                            ADD   #ERRINT-.,R0        ;HANDLER
000040' 005001                                    CLR   R1                  ;FILL ALL UNUSED VECTORS WITH TRAP...
000042' 010021                            10$:    MOV   R0,(R1)+            ;HANDLER
000044' 012721  000340                            MOV   #PRI07,(R1)+
000050' 020127  001000                            CMP   R1,#1000
000054' 002772                                    BLT   10$

000056' 010700                                    MOV   PC,R0               ;SETUP PARITY TRAP VECTOR
000060' 062700  000404                            ADD   #PARINT-.,R0
000064' 010037  000140                            MOV   R0,@#PARVEC
000070' 012737  000340  000142                    MOV   #PRI07,@#PARVEC+2

000076' 010700                                    MOV   PC,R0               ;SETUP DMA INTERRUPT VECTOR
000100' 062700  000306                            ADD   #DMAINT-.,R0
000104' 010037  000114                            MOV   R0,@#DMAVEC
000110' 012737  000300  000116                    MOV   #PRI06,@#DMAVEC+2

000116' 010700                                    MOV   PC,R0               ;SETUP CSR WRITE VECTOR
000120' 062700  000232                            ADD   #CSRWRT-.,R0
000124' 010037  000064                            MOV   R0,@#CSRVEC
000130' 012737  000200  000066                    MOV   #PRI04,@#CSRVEC+2

000136' 010700                                    MOV   PC,R0               ;SETUP SANTITY TIMER VECTOR
000140' 062700  000240                            ADD   #TIMINT-.,R0
000144' 010037  000134                            MOV   R0,@#SANVEC
000150' 012737  000240  000136                    MOV   #PRI05,@#SANVEC+2

000156' 013700  021040                            MOV   @#PCSRSW,R0         ;GET SWITCH PACK BITS
000162' 052700  176000                            BIS   #176000,R0          ;MAP THEM INTO HOST I/O PAGE
000166' 006300                                    ASL   R0                  ;SHIFT OVER TO POSITION CORRECTLY
000170' 006300                                    ASL   R0
000172' 006300                                    ASL   R0
```

```
000174' 062700 000004                  ADD     #4,R0                   ;PCSR2 IS PCSR0+4
000200' 010067 000324                  MOV     R0,IPCSR2               ;SAVE PCSR2 ADDRESS
000204' 012767 000003 000320           MOV     #3,IPCSR2+2             ;HIGH ORDER BITS 17:16
000212' 005067 000304                  CLR     FLG3                    ;INITIALIZE FLAG WORD
000216' 012737 004000 021000           MOV     #DN1,@#IPCSR0           ;TELL HOST THE LOAD AND START FINISHED
000224' 106427 000000          15$:    MTPS    #PRI00                  ;ALLOW INTERRUPTS
000230' 005767 000266          20$:    TST     FLG3                    ;WAIT FOR AN INTERRUPT
000234' 001775                         BEQ     20$
000236' 106427 000340                  MTPS    #PRI07                  ;RAISE CPU PRIORITY
000242' 032767 000001 000252           BIT     #CSRFLG,FLG3            ;DID HOST GIVE US A COMMAND?
000250' 001001                         BNE     30$                     ;YES
000252' 000777                         BR      .                       ;NO, ERROR SO JUST SIT HERE...
                                                                       ;FOR LACK OF ANYTHING BETTER TO DO
000254' 113700 021000          30$:    MOVB    @#IPCSR0,R0             ;GET WHAT HOST WROTE TO PCSR0
000260' 042700 177760                  BIC     #177760,R0              ;STRIP ALL BUT COMMAND BITS
000264' 022700 000017                  CMP     #17,R0                  ;RETURN TO OPERATIONAL MICROCODE?
000270' 001425                         BEQ     60$                     ;YES
000272' 010701                         MOV     PC,R1                   ;GET ADDRESS OF OUR COMMAND TABLE
000274' 062701 000206                  ADD     #TBLC-.,R1
000300' 006300                         ASL     R0                      ;MAKE COMMAND A BYTE OFFSET
000302' 060001                         ADD     R0,R1                   ;USE IT TO INDEX INTO COMMAND TABLE
000304' 061101                         ADD     (R1),R1                 ;R1 NOW HAS COMMAND ROUTINE ADDRESS
000306' 004711                         JSR     PC,(R1)                 ;EXECUTE AS COMMANDED FROM HOST
000310' 103404                         BCS     40$                     ;ERROR OCCURRED
000312' 112767 000001 000206           MOVB    #INMON,PCSR1            ;INDICATE TO HOST WE ARE BACK IN...
000320' 000403                         BR      45$                     ;MICROMONITR
000322' 112767 000003 000176   40$:    MOVB    #INERR,PCSR1            ;INDICATE TO HOST ERROR OCCURRED
000330' 016737 000172 021020   45$:    MOV     PCSR1,@#IPCSR1
000336' 005067 000160                  CLR     FLG3                    ;RESET FLAG WORD
000342' 000730                         BR      15$                     ;GO WAIT FOR ANOTHER COMMAND
000344' 005000                 60$:    CLR     R0                      ;FAKE SUCCESSFUL SELF TEST RESULTS
000346' 000137 040006                  JMP     @#40006                 ;START OPERATIONAL MICROCODE
000352' 052767 000001 000142   CSRWRT: BIS     #CSRFLG,FLG3            ;INDICATE A CSR WRITE INTERRUPT OCCURED
000360' 000002                         RTI
000362' 052767 000002 000132   ERRINT: BIS     #ERRFLG,FLG3            ;INDICATE A UNEXPECTED INTERRUPT OCCURED
000370' 012737 020000 021000           MOV     #UNIERR,@#IPCSR0        ;TELL HOST AN UNEXPECTED INTERRUPT
                                                                       ;HAPPENED
000376' 000777                         BR      .                       ;JUST SIT HERE AND SPIN WHEELS
                                                                       ;COUNT ON THE HOST TO TIMEOUT WAITING
000400' 005267 000120          TIMINT: INC     SANTIM                  ;COUNT TICKS AS THEY OCCUR
000404' 000002                         RTI
000406' 013767 021002 000124   DMAINT: MOV     @#DMACSR,DMDONE         ;GET DMA STATUS
000414' 032767 040000 000116           BIT     #BIT14,DMDONE           ;DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
000422' 001404                         BEQ     10$                     ;NO
000424' 012737 040000 021000           MOV     #NXMERR,@#IPCSR0        ;YES, TELL HOST A NON-EXISTANT MEMORY
                                                                       ;LOCATION WAS ADDRESSED
000432' 000407                         BR      20$
000434' 032767 100000 000076   10$:    BIT     #BIT15,DMDONE           ;DID A NPR TIMEOUT OCCUR?
```

```
000442' 001407                               BEQ     30$                       ;NO
000444' 012737  100000  021000               MOV     #NPRERR,@#IPCSR0          ;TELL HOST NPR TIMEOUT HAPPENED
000452' 012737  100000  021002   20$:        MOV     #BIT15,@#DMACSR           ;CLEAR THE INTERRUPT IN THE DMA ENGINE
000460' 000777                               BR      .                         ;SIT HERE AND SPIN WHEELS
000462' 000002                   30$:        RTI


000464' 052767  000004  000030   PARINT: BIS     #PARFLG,FLG3              ;SET PARITY ERROR OCCURRED
000472' 012737  010000  021000               MOV     #PARERR,@#IPCSR0          ;TELL HOST A LINK MEMORY PARITY ERROR
                                                                               ;OCCURRED
000500' 000002                               RTI

000502' 000042                   TBLC:   .WORD   CLRC-.                    ;CLEAR COMMAND
000504' 000050                           .WORD   MICC1-.                   ;PCSR0 INTERRUPT BIT TEST
000506' 000136                           .WORD   MICC2-.                   ;SANITY TIMER TEST
000510' 000222                           .WORD   MICC3-.                   ;COMPREHENSIVE LINK MEMORY TEST
000512' 001662                           .WORD   MICC4-.                   ;DMA 'TO' ADDRESS REGISTER TEST
000514' 001772                           .WORD   MICC5-.                   ;DMA 'FROM' ADDRESS REGISTER TEST
000516' 002166                           .WORD   MICC6-.                   ;DMA BLOCK TRANSFER TEST
000520' 002346                           .WORD   MICC7-.                   ;RIPPLE TEST

000522' 000000                   FLG3:   .WORD   0                         ;FLAG WORD
000524' 000000                   SANTIM: .WORD   0                         ;COUNT FOR SANITY TIMER
000526' 000000                   PCSR1:  .WORD   0                         ;COPY OF WHAT GOES TO IPCSR1
000530' 000000  000000           IPCSR2: .WORD   0,0                       ;ADDRESS IN HOST MEMORY FOR PCSR2
000534' 000000  000000           PCBADR: .WORD   0,0                       ;ADDRESS IN HOST MEMORY FOR PCB
000540' 000000                   DMDONE: .WORD   0                         ;DMA INTERRUPT HAPPENED
000542' 000377                   BAKPAT: .WORD   377                       ;BACKGROUND PATTERN FOR MEMORY TESTS
```

```



                              .SBTTL  MODULE C, MICRO CLEAR FUNCTION


000544' 105067  177757        CLRC:   CLRB    PCSR1+1                 ;CLEAR THE SELF TEST FIELD
000550' 000241                        CLC
000552' 000207                        RTS     PC
```

```
                                       .SBTTL  MODULE C, MICROTEST #1

000554' 112767  000002  177744  MICC1:  MOVB    #INTST,PCSR1
000562' 016737  177740  021020          MOV     PCSR1,@#IPCSR1
000570' 016737  177734  021010          MOV     IPCSR2,@#MDMA0  ;PICK UP ADDRESS OF PCBB
000576' 016737  177730  021012          MOV     IPCSR2+2,@#MDMA1
000604' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 HAS CONTENTS OF HOST'S PCSR2
000610' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 HAS CONTENTS OF HOST'S PCSR3
000614' 010037  021010                  MOV     R0,@#MDMA0              ;FETCH CONTENTS OF PCBB+0
000620' 010137  021012                  MOV     R1,@#MDMA1

                                ;AT THIS POINT MDMAR0 WILL CONTAIN THE INTERRUPT BIT THAT WE ARE TO SET

000624' 013737  021014  021000          MOV     @#MDMAR0,@#IPCSR0       ;SET CORRESPONDING INTERRUPT BIT
000632' 112767  000001  177667          MOVB    #1,PCSR1+1              ;TELL HOST WHAT TEST WE JUST FINISHED
000640' 000241                          CLC                             ;SUCCESS
000642' 000207                          RTS     PC
```

```
                                            .SBTTL  MODULE C, MICROTEST #2


000644' 112767  000002  177654  MICC2:  MOVB    #!NTST,PCSR1            ;TELL HOST WE ARE TESTING
000652' 016737  177650  021020          MOV     PCSR1,@#IPCSR1
000660' 005067  177640                  CLR     SANTIM                  ;CLEAR TIMER COUNT
000664' 106427  000200                  MTPS    #PRI04                  ;LOWER PRIORITY TO ALLOW TIMER INTERRUPT
000670' 026727  177630  000012  10$:    CMP     SANTIM,#10.             ;WAIT FOR TEN TICKS OF THE TIMER
000676' 002774                          BLT     10$
000700' 106427  000340                  MTPS    #PRI07                  ;RAISE PRIORITY TO STOP TIMER
000704' 112767  000002  177615          MOVB    #2,PCSR1+1              ;TELL HOST WHAT TEST WE JUST FINISHED
000712' 016737  177610  021020          MOV     PCSR1,@#IPCSR1
000720' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST WE ARE DONE
000726' 000241                          CLC
000730' 000207                          RTS     PC
```

```
                                 .SBTTL  MODULE C, MICROTEST #3

                                 ;*********************************************************************8
                                 ;
                                 ;THIS MICROTEST CHECKS THE LINK MEMORY. IT DOES SO BY RUNNING A SERIES
                                 ;OF MICROSUBTESTS ON THE MEMORY BETWEEN 16K AND 32K (100000-177776).
                                 ;THE SUBTESTS ARE:
                                 ;       A-ACCESS TEST
                                 ;       B-ADDRESS SHIFT TEST
                                 ;       C-DATA LATCH TEST
                                 ;       D-ADDRESS BIT SHIFT #1
                                 ;       E-ADDRESS BIT SHIFT #2
                                 ;       F-MARCH TEST
                                 ;
                                 ;*********************************************************************8

000732' 112767 000002 177566  MICC3:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
000740' 016737 177562 021020          MOV     PCSR1,@#IPCSR1
000746' 010700                        MOV     PC,R0                   ;GET ADDRESS OF SUBTEST LIST
000750' 062700 000134                 ADD     #C3STBL-.,R0

000754' 011001                10$:    MOV     (R0),R1                 ;GET OFFSET FROM ENTRY TO SUBTEST
000756' 001437                        BEQ     30$                     ;NO MORE ENTRIES
000760' 060001                        ADD     R0,R1                   ;CALC ACTUAL SUBTEST ADDRESS
000762' 004711                        JSR     PC,(R1)                 ;GO EXECUTE SUBTEST
000764' 103403                        BCS     20$                     ;ERROR OCCURRED IN SUBTEST
000766' 062700 000002                 ADD     #2,R0                   ;POINT TO NEXT SUBTEST IN LIST
000772' 000770                        BR      10$

000774' 016737 177530 021010  20$:    MOV     IPCSR2,@#MDMA0          ;GET CONTENTS OF HOST'S PCSR2 AND PCSR3
001002' 016737 177524 021012          MOV     IPCSR2+2,@#MDMA1
001010' 013700 021014                 MOV     @#MDMAR0,R0             ;R0=CONTENTS OF PCSR2
001014' 013702 021014                 MOV     @#MDMAR0,R2             ;R2=CONTENTS OF PCSR3
001020' 010037 021010                 MOV     R0,@#MDMA0              ;POINT TO PCBB+0
001024' 010237 021012                 MOV     R2,@#MDMA1
001030' 016737 000066 021026          MOV     SUBNUM,@#MDMAW0         ;LOAD PCBB+0 WITH SUBTEST #...
                                                                      ;AND PCBB+1 WITH ERROR TYPE
001036' 010137 021026                 MOV     R1,@#MDMAW0             ;LOAD PCBB+2 WITH FAILING ADDRESS
001042' 010337 021026                 MOV     R3,@#MDMAW0             ;LOAD PCBB+4 WITH GOOD DATA
001046' 010437 021026                 MOV     R4,@#MDMAW0             ;LOAD PCBB+6 WITH BAD DATA
001052' 000261                        SEC                             ;TELL MICROMONITOR THIS TEST FAILED
001054' 000401                        BR      40$

001056' 000241                30$:    CLC                             ;TELL MICROMONITOR THIS TEST WAS SUCCESSFUL
001060' 112767 000003 177441  40$:    MOVB    #3,PCSR1+1              ;TELL HOST WHICH TEST THIS WAS
001066' 016737 177434 021020          MOV     PCSR1,@#IPCSR1
001074' 012737 004000 021000          MOV     #DNI,@#IPCSR0           ;TELL HOST THIS MICROTEST IS FINISHED
001102' 000207                        RTS     PC

001104' 000020                C3STBL: MICC3A-.
001106' 000160                        MICC3B-.
001110' 000322                        MICC3C-.
001112' 000436                        MICC3D-.
001114' 000704                        MICC3E-.
001116' 001024                        MICC3F-.
001120' 000000                        .WORD   0                       ;END OF LIST
```

```
342
343 001122'     000          SUBNUM: .BYTE   0                 ;CURRENT SUBTEST # BEING EXECUTED
344 001123'     000          ERRTYP: .BYTE   0                 ;TYPE OF ERROR, DATA =0, ADDRESS=1
345
```

```
                                        .SBTTL  MODULE C, MICROTEST #3, MICROSUBTEST A

                                        ;*****************************************************************************
                                        ;
                                        ;THIS IS AN ACCESS TEST OF LINK MEMORY. IT WRITES DATA
                                        ;ALL THROUGH 16K TO 32K(100000,177772) AND VERIFIES SAME. IT THEN WRITES
                                        ;ZEROS . IT ALSO CHECKS FOR PARITY ERRORS.
                                        ;
                                        ;*****************************************************************************

001124' 010046                          MICC3A: MOV     R0,-(SP)                ;SAVE R0
001126' 112767  000001  177766                  MOVB    #1,SUBNUM               ;TELL WE ARE IN SUBTEST A
001134' 106427  000300                          MTPS    #PRI06                  ;ALLOW PARITY ERRORS

001140' 010600                                  MOV     SP,R0                   ;SAVE THE STACK POINTER FOR A MOMENT
001142' 012706  177774                          MOV     #LINADR+LINSIZ,SP       ;POINT STACK TO TOP OF LINK MEMORY
001146' 012705  037776                          MOV     #LINSIZ/2,R5            ;NUMBER OF WORDS IN LINK MEMORY
001152' 004767  000000                  5$:     JSR     PC,10$                  ;WRITE ADDRESS OF NEXT INSTUCTION
001156' 077503                          10$:    SOB     R5,5$                   ;ALL OVER LINK MEMORY

001160' 010006                                  MOV     R0,SP                   ;RESTORE STACK
001162' 010703                                  MOV     PC,R3                   ;GET DATA THAT WAS WRITTEN TO LINK MEMORY
001164' 062703  177772                          ADD     #10$-.,R3

001170' 012701  100000                          MOV     #LINADR,R1              ;POINT TO BASE OF LINK MEMORY
001174' 011104                          12$:    MOV     (R1),R4                 ;READ DATA
001176' 032767  000004  177316                  BIT     #PARFLG,FLG3            ;DID A PARITY ERROR OCCUR?
001204' 001002                                  BNE     20$                     ;YES

001206' 020304                          15$:    CMP     R3,R4                   ;NO, WAS DATA GOOD?
001210' 001410                                  BEQ     30$                     ;YES
001212' 112767  000003  177306          20$:    MOVB    #INERR,PCSR1            ;TELL HOST THIS SUBTEST FAILED
001220' 112767  000000  177675                  MOVB    #DATERR,ERRTYP          ;TELL MICROMONITOR TYPE OF ERROR
001226' 000261                                  SEC                             ;TELL MICROMONITOR ERROR OCCURRED
001230' 000414                                  BR      40$

001232' 062701  000002                  30$:    ADD     #2,R1                   ;POINT TO NEXT ADDRESS
001236' 020127  177774                          CMP     R1,#LINADR+LINSIZ       ;DONE ALL OF LINK MEMORY?
001242' 001354                                  BNE     12$                     ;NOT YET

001244' 012701  100000                          MOV     #LINADR,R1              ;OK NOW FILL WITH ZEROES
001250' 005021                          32$:    CLR     (R1)+
001252' 020127  177774                          CMP     R1,#LINADR+LINSIZ
001256' 001374                                  BNE     32$

001260' 000241                          35$:    CLC                             ;INDICATE SUCCESS
001262' 012600                          40$:    MOV     (SP)+,R0                ;RESTORE R0
001264' 000207                                  RTS     PC
```

```
                                          .SBTTL  MODULE C, MICROTEST #3, MICROSUBTEST B

                                          ;*********************************************************************
                                          ;
                                          ;ADDRESS SHIFT TEST
                                          ;
                                          ;THIS TEST ASSUMES ALL MEMORY BETWEEN 100000 AND 177776 IS ZEROS
                                          ;IT CHECKS FOR PROPER BANK SELECTION BY WRITING 1'S IN A LOCATION
                                          ;AND CHECKING FOR 1'S IN THE SAME LOCATION OF OTHER 4K BANKS (ADDRESS ERROR).
                                          ;IT ALSO CHECKS FOR DATA ERRORS I.E. NON-ZERO DATA IN LOCATIONS NOT
                                          ;WRITTEN WITH 1'S AND MAKES SURE LOCATIONS WRITTEN WITH 1'S HAVE 1'S IN THEM
                                          ;
                                          ;*********************************************************************

 001266' 010046                           MICC3B: MOV     R0,-(SP)                      ;SAVE R0
 001270' 112767 000002 177624                     MOVB    #2,SUBNUM                     ;TELL HIM WE ARE IN THE SECOND SUBTEST

 001276' 012700 100000                            MOV     #LINADR,R0                    ;SET 'WRITTEN TO' ADDRESS
 001302' 012701 100000                    10$:    MOV     #LINADR,R1                    ;SET 'READ FROM' ADDERSS
 001306' 012710 177777                            MOV     #177777,(R0)                  ;WRITE DATA INTO LOCATION AT BASE...
                                                                                        ;OF A 4K BOUNDARY
 001312' 011104                           20$:    MOV     (R1),R4                       ;READ DATA FROM MEMORY ADDRESS WHICH...
                                                                                        ;IS AN EVEN 4K INCREMENT AWAY
 001314' 020001                                   CMP     R0,R1                         ;IS 'READ FROM' AND 'WRITTEN TO'
                                                                                        ;ADDRESSES THE SAME?
 001316' 001416                                   BEQ     70$                           ;YES, SO DATA IN BOTH SHOULD BE THE SAME
 001320' 005003                                   CLR     R3                            ;GOOD DATA IS ZEROS
 001322' 005704                                   TST     R4                            ;DATA IN 'READ FROM' MUST BE 0'S
 001324' 001430                                   BEQ     50$                           ;OK GO CHANGE 'READ FROM' ADDRESS
                                                                                        ;ERROR OCCURRED BUT WE DON'T KNOW IF...
                                                                                        ;IT IS A DATA ERROR OR AN ADDRESS ERROR
 001326' 022704 177777                            CMP     #177777,R4                    ;WAS DATA READ ALL 1'S?
 001332' 001015                                   BNE     40$                           ;NO, SO IT WAS A DATA ERROR
 001334' 112767 000001 177561                     MOVB    #ADRERR,ERRTYP                ;YES, IT WAS AN ADDRESS ERROR
 001342' 112767 000003 177156                     MOVB    #INERR,PCSR1                  ;TELL HOST THIS TEST FAILED
 001350' 000261                                   SEC                                   ;INDICATE FAILURE
 001352' 000425                                   BR      60$                           ;LEAVE THIS SUBTEST
 001354' 022704 177777                    70$:    CMP     #177777,R4                    ;DATA READ MUST BE ALL 1'S
 001360' 001412                                   BEQ     50$                           ;IT IS GOOD
 001362' 012703 177777                            MOV     #177777,R3                    ;GOOD DATA IS ONES
 001366' 112767 000000 177527             40$:    MOVB    #DATERR,ERRTYP                ;IT WAS A DATA ERROR
 001374' 112767 000003 177124                     MOVB    #INERR,PCSR1                  ;TELL HOST THIS TEST FAILED
 001402' 000261                                   SEC                                   ;INDICATE FAILURE
 001404' 000410                                   BR      60$                           ;LEAVE THIS SUBTEST

 001406' 062701 020000                    50$:    ADD     #20000,R1                     ;CHANGE 'READ FROM' ADDRESS BY 4K
 001412' 103337                                   BCC     20$                           ;CONTINUE SAME 'WRITTEN TO' IF NOT PAST 32K
 001414' 005010                                   CLR     (R0)                          ;CLEAR OLD 'WRITTEN TO' ADDRESS
 001416' 062700 020000                            ADD     #20000,R0                     ;CHANGE 'WRITTEN TO' ADDRESS BY 4K
 001422' 103327                                   BCC     10$                           ;CONTINUE IF NOT PAST 32K
 001424' 000241                                   CLC                                   ;INDICATE SUCCESS OF THIS SUBTEST
 001426' 012600                           60$:    MOV     (SP)+,R0                      ;RESTORE R0
 001430' 000207                                   RTS     PC
```

```
                                        .SBTTL  MODULE C, MICROTEST #3, MICROSUBTEST C

                                        ;************************************************************************
                                        ;
                                        ;DATA LATCH TEST
                                        ;
                                        ;AT THE FIRST LOCATION OF EACH 4K BANK A '1' IS SHIFTED THROUGH EACH
                                        ;BIT POSITION AND CHECKED FROM LSB TO MSB.
                                        ;THEN IN THE SAME LOCATION A '0' IS SHIFTED THROUGH EACH BIT POSITION
                                        ;AND CHECKED.
                                        ;
                                        ;************************************************************************


001432' 010046                          MICC3C: MOV     R0,-(SP)                        ;SAVE R0
001434' 112767  000003  177460                  MOVB    #3,SUBNUM                       ;TELL WEA ARE IN SUBTEST 'C'
001442' 012701  100000                          MOV     #LINADR,R1                      ;GET BASE ADDRESS OF LINK MEMORY
001446' 012700  000001          1$:             MOV     #1,R0                           ;DATA = 1 IN LEAST SIGNIFICANT BIT
001452' 010002                                  MOV     R0,R2                           ;INDICATE WE ARE SHIFTING A '1'
001454' 010011                  2$:             MOV     R0,(R1)                         ;WRITE LOCATION WITH GOOD DATA
001456' 011104                                  MOV     (R1),R4                         ;READ DATA FROM SAME LOCATION
001460' 020004                  3$:             CMP     R0,R4                           ;IS DATA THE SAME AS WRITTEN?
001462' 001411                                  BEQ     4$                              ;YES, OK
001464' 112767  000000  177431                  MOVB    #DATERR,ERRTYP                  ;ERROR IS DATA ERROR
001472' 010003                                  MOV     R0,R3                           ;GOOD DATA
001474' 112767  000003  177024                  MOVB    #INERR,PCSR1                    ;TELL HOST THIS TEST FAILED
001502' 000261                                  SEC                                     ;INDICATE THIS SUBTEST FAILED
001504' 000417                                  BR      6$                              ;LEAVE THIS SUBTEST
001506' 005702                  4$:             TST     R2                              ;ARE WE SHIFTING A 1 OR A 0?
001510' 001406                                  BEQ     5$                              ;ZERO
001512' 006300                                  ASL     R0                              ;SHIFT THE ONE OVER
001514' 103357                                  BCC     2$                              ;IF THE '1' HAS NOT BEEN SHIFTED...
                                                                                        ;THRU THE 16 POSITIONS CONTINUE WITH 1
001516' 005002                                  CLR     R2                              ;ELSE START SHIFTING A '0'
001520' 012700  177776                          MOV     #177776,R0                      ;START WITH LSB = 0 ALL OTHERS 1'S
001524' 000753                                  BR      2$                              ;CONTINUE WITH SHIFTING A 0
001526' 000261                  5$:             SEC                                     ;MOVE '0' OVER ONE BIT POSITION
001530' 006100                                  ROL     R0                              ;HAS '0' BEEN IN ALL POSITIONS?
001532' 103750                                  BCS     2$                              ;NO CONTINUE WITH SHIFTING A 0
001534' 062701  020000                          ADD     #SIZ4K,R1                       ;CONTINUE TEST AT NEXT 4K BOUNDARY...
001540' 103342                                  BCC     1$                              ;IF NOT PAST 32K
001542' 000241                                  CLC                                     ;INDICATE THIS SUBTEST SUCCESSFUL
001544' 012600                  6$:             MOV     (SP)+,R0                        ;RESTORE R0
001546' 000207                                  RTS     PC
```

```
                                        .SBTTL  MODULE C, MICROTEST #2, MICROSUBTEST D

                                        ;*****************************************************************
                                        ;
                                        ;ADDRESS BIT SHIFT #1
                                        ;
                                        ;THIS TEST CHECKS FOR DUAL ADDRESS PROBLEMS BY FIRST WRITING A BACKGROUND
                                        ;PATTERN THROUGHOUT LINK MEMORY.
                                        ;THEN STARTING AT THE LOWEST LOCATION IN A BANK IT WRITES THE COMPLEMENT
                                        ;OF THE BACKGROUND PATTERN. THEN READS THE MEMORY FOR CORRECT CONTENTS
                                        ;IT THEN SHIFTS A '1' THROUGH THE ADDRESS POINTER AND REPEATS THE ABOVE.
                                        ;IT DOES THIS FOR EACH 4K BANK.
                                        ;
                                        ;*****************************************************************

001550' 010046                          MICC3D: MOV     R0,-(SP)                ;SAVE R0
001552' 112767  000004  177342                  MOVB    #4,SUBNUM               ;TELL WE ARE IN SUBTEST 4
001560' 012767  000377  176754                  MOV     #377,BAKPAT             ;LOAD BAKPAT CONSTANT
001566' 005003                          1$:     CLR     R3                      ;CONTAINS 4K BANK ADDRESS
001570' 012700  100000                  2$:     MOV     #LINADR,R0              ;WRITE LINK MEMORY WITH BACKGROUND...
001574' 016720  176742                  3$:     MOV     BAKPAT,(R0)+            ;PATTERN
001600' 020027  177774                          CMP     R0,#LINADR+LINSIZ
001604' 103773                                  BLO     3$
001606' 016700  176730                          MOV     BAKPAT,R0               ;R0 CONTAINS GOOD DATA

001612' 005002                          4$:     CLR     R2                      ;R2 WILL BE OUR 'WRITTEN TO' ADDRESS
001614' 050302                          6$:     BIS     R3,R2                   ;INDEX INTO THIS 4K BANK
001616' 020227  100000                          CMP     R2,#LINADR              ;IS RESULT LESS THAN LINK MEM BASE?
001622' 103453                                  BLO     16$                     ;YES, DON'T USE THIS ADDRESS
001624' 020227  177774                          CMP     R2,#LINADR+LINSIZ       ;IS RESULT LARGER THAN LINK MEM TOP?
001630' 103065                                  BHIS    20$                     ;YES, DON'T USE THIS ADDRESS EITHER
001632' 000312                                  SWAB    (R2)                    ;WRITE COMPLEMENT OF PATTERN
001634' 005001                                  CLR     R1                      ;R1 WILL BE OUR 'READ FROM' ADDRESS
001636' 050301                          7$:     BIS     R3,R1                   ;INDEX INTO THIS 4K BANK
001640' 020127  100000                          CMP     R1,#LINADR              ;IS RESULT LESS THAN LINK MEM BASE?
001644' 103431                                  BLO     12$                     ;YES, DON'T USE THIS ADDRESS
001646' 020127  177774                          CMP     R1,#LINADR+LINSIZ       ;IS RESULT LARGER THAN LINK MEM TOP?
001652' 103036                                  BHIS    15$                     ;YES, DON'T USE THIS ADDRESS EITHER
001654' 011104                                  MOV     (R1),R4                 ;READ DATA
001656' 020102                                  CMP     R1,R2                   ;IS 'READ FROM' AND 'WRITTEN TO' SAME?
001660' 001417                                  BEQ     10$                     ;YES, GO CHECK DATA
001662' 020004                                  CMP     R0,R4                   ;NO, DATA READ SHOULD BE SAME AS BAKPAT
001664' 001421                                  BEQ     12$                     ;IF SO CONTINUE WITH NEW INDEX

001666' 112767  000001  177227          8$:     MOVB    #ADRERR,ERRTYP          ;INDICATE ADDRESS ERROR
001674' 010003                                  MOV     R0,R3                   ;GET GOOD DATA
001676' 000261                                  SEC                             ;INDICATE THIS SUBTEST FAILED
001700' 112767  000003  176620                  MOVB    #INERR,PCSR1            ;TELL HOST THIS TEST FAILED
001706' 000442                                  BR      25$                     ;LEAVE THIS SUBTEST
001710' 020067  176626                          CMP     R0,BAKPAT               ;DOES R0 CONTAIN SWAPPED DATA?
001714' 001405                                  BEQ     12$                     ;NO
001716' 000403                                  BR      11$                     ;YES

001720' 000300                          10$:    SWAB    R0                      ;MAKE GOOD DATA LIKE SWAPPED BAKPAT
001722' 020004                                  CMP     R0,R4                   ;IS DATA READ SAME AS DATA WRITTEN?
001724' 001360                                  BNE     8$                      ;NO, ERROR
```

```
001726' 000300                  11$:    SWAB    R0                      ;RESTORE GOOD DATA TO BAKPAT
001730' 040301                  12$:    BIC     R3,R1                   ;REMOVE 4K BANK SELECT FROM 'READ FROM'
001732' 005701                          TST     R1                      ;DID AN INDEX EXIST?
001734' 001001                          BNE     13$                     ;YES
001736' 005201                          INC     R1                      ;NO, MAKE ONE
001740' 006101                  13$:    ROL     R1                      ;SHIFT INDEX OVER
001742' 020127  020000                  CMP     R1,#SIZ4K               ;IS IT LESS THAN 4K
001746' 103733                          BLO     7$                      ;YES, CONTINUE CHECKING FOR BOGUS DATA
                                                                        ;IN THIS 4K BANK
001750' 000312                  15$:    SWAB    (R2)                    ;RESTORE CONTENTS OF 'WRITTEN TO'
001752' 040302                  16$:    BIC     R3,R2                   ;REMOVE BANK SELECT FROM 'WRITTEN TO'
001754' 005702                          TST     R2                      ;DOES 'WRITTEN TO' CONTAIN AN ADDRESS?
001756' 001001                          BNE     18$                     ;YES
001760' 005202                          INC     R2                      ;NO, MAKE ONE
001762' 006102                  18$:    ROL     R2                      ;SHIFT 'WRITTEN TO' ADDRESS
001764' 020227  020000                  CMP     R2,#SIZ4K               ;IS 'WRITTEN TO' LARGER THAN 4K?
001770' 103711                          BLO     6$                      ;NO
001772' 060203                          ADD     R2,R3                   ;MAKE 'WRITTEN TO' POINT TO NEXT 4K...
001774' 103403                          BCS     20$                     ;IF IT HAS WRAPPED AROUND DO COMPLEMENTED
001776' 020327  177774                  CMP     R3,#LINADR+LINSIZ       ;IS RESULT LARGER THAN LINK MEMORY TOP?
002002' 103703                          BLO     4$                      ;NO
002004' 000367  176532          20$:    SWAB    BAKPAT                  ;REPEAT TEST WITH COMPLEMENTED PATTERN
002010' 001666                          BEQ     1$
002012' 000241                          CLC                             ;INDICATE THIS SUBTEST SUCCESS
002014' 012600                  25$:    MOV     (SP)+,R0                ;RESTORE R0
002016' 000207                          RTS     PC
```

```
                                        .SBTTL  MODULE C, MICROTEST #3, MICROSUBTEST E

                                        ;*************************************************************************
                                        ;
                                        ;ADDRESS BIT SHIFT #2
                                        ;
                                        ;THIS TEST CHECKS FOR DUAL ADDRESSING PROBLEMS BY WRITING EACH LOCATION
                                        ;OF LINK MEMORY WITH ITS ADDRESS AND THEN VERIFYING EACH LOCATION.
                                        ;IT THEN DOES THE SAME THING BUT USES THE COMPLEMENT OF THE ADDRESS
                                        ;AS DATA.
                                        ;
                                        ;*************************************************************************

002020' 010046                          MICC3E: MOV     R0,-(SP)                ;SAVE R0
002022' 112767  000005  177072                  MOVB    #5,SUBNUM               ;TELL WHICH SUBTEST WE ARE IN
002030' 005003                                  CLR     R3                      ;FLAG INDICATING ADDRESS IS COMPLEMENTED
002032' 012701  100000                          MOV     #LINADR,R1              ;GET STARTING ADDRESS OF LINK MEMORY
002036' 010100                          1$:     MOV     R1,R0                   ;GET ADDRESS TO WORK WITH
002040' 005703                                  TST     R3                      ;SHOULD WE COMPLEMENT THE DATA TO STORE?
002042' 001401                                  BEQ     2$                      ;NO STORE AS IS
002044' 005100                                  COM     R0                      ;COMPLEMENT DATA
002046' 010021                          2$:     MOV     R0,(R1)+                ;WRITE DATA
002050' 020127  177774                          CMP     R1,#LINADR+LINSIZ       ;IS NEW ADDRESS LARGER THAN LINK MEM?
002054' 103770                                  BLO     1$                      ;NO, KEEP FILLING LINK MEMORY

002056' 014104                          3$:     MOV     -(R1),R4                ;READ DATA STORED
002060' 020004                                  CMP     R0,R4                   ;IS DATA READ SAME AS WRITTEN?
002062' 001411                                  BEQ     4$                      ;YES
002064' 112767  000001  177031                  MOVB    #ADRERR,ERRTYP          ;INDICATE ADDRESS ERROR
002072' 010003                                  MOV     R0,R3                   ;GET GOOD DATA
002074' 000261                                  SEC                             ;INDICATE FAILURE
002076' 112767  000003  176422                  MOVB    #INERR,PCSR1            ;TELL HOST THIS TEST FAILED
002104' 000414                                  BR      10$                     ;LEAVE THIS SUBTEST
002106' 010100                          4$:     MOV     R1,R0                   ;CALC GOOD DATA FOR NEXT LOCATION
002110' 162700  000002                          SUB     #2,R0
002114' 005703                                  TST     R3
002116' 001401                                  BEQ     5$
002120' 005100                                  COM     R0
002122' 020127  100000                  5$:     CMP     R1,#LINADR              ;HAVE WE CHECKED ALL LOCATIONS
002126' 101353                                  BHI     3$                      ;NOT YET
002130' 005103                                  COM     R3                      ;HAVE WE DONE IT COMPLEMENTED?
002132' 001341                                  BNE     1$                      ;NO, REPEAT WITH COMPLEMENT
002134' 000241                                  CLC                             ;INDICATE SUCCESS
002136' 012600                          10$:    MOV     (SP)+,R0                ;RESTORE R0
002140' 000207                                  RTS     PC
```

```
                                        .SBTTL  MODULE C, MICROTEST #3, MICROSUBTEST F

                                        ;*************************************************************************
                                        ;
                                        ;MARCH TEST
                                        ;
                                        ;THIS TEST WRITES A BACKGROUND PATTERN IN ALL OF LINK MEMORY
                                        ;
                                        ;1-READ EVERY LOCATION FOR CORRECT DATA, SWAPS BYTES AT EACH LOCATION
                                        ;AND PROCEED IN MAX TO MIN DIRECTION
                                        ;2-READ EVERY LOCATION FOR SWAPPED DATA, WRITES ORIGINAL PATTERN IN EACH
                                        ;LOCATION AND PROCEEDS IN MIN TO MAX DIRECTION
                                        ;3-REPEAT 1 GOING IN MIN TO MAX DIRECTION
                                        ;4-REPEAT 2 GOING IN MAX TO MIN DIRECTION
                                        ;
                                        ;*************************************************************************

002142' 010046                          MICC3F: MOV     R0,-(SP)                        ;SAVE R0
002144' 112767  000006  176750                  MOVB    #6,SUBNUM                       ;TELL WHICH SUBTEST WE ARE IN
002152' 005003                                  CLR     R3                              ;ADDRESS DIRECTION FLAG 0 = MIN.->MAX
002154' 012701  100000                  10$:    MOV     #LINADR,R1                      ;FILL LINK MEMORY WITH BACKGROUND PATTERN
002160' 012700  000377                          MOV     #377,R0                         ;BACKGROUND PATTERN=LOW BYTE ALL 1'S
002164' 010021                          12$:    MOV     R0,(R1)+
002166' 020127  177774                          CMP     R1,#LINADR+LINSIZ
002172' 103774                                  BLO     12$

002174' 014104                          20$:    MOV     -(R1),R4                        ;STARTING FROM THE TOP, READ DATA
002176' 020004                                  CMP     R0,R4                           ;R0 = GOOD DATA, R4 = DATA READ
002200' 001411                                  BEQ     30$                             ;IF SAME OK
002202' 112767  000000  176713                  MOVB    #DATERR,ERRTYP                  ;INDICATE DATA ERROR
002210' 010003                                  MOV     R0,R3                           ;GET GOOD DATA
002212' 000261                                  SEC                                     ;INDICATE FAILURE
002214' 112767  000003  176304                  MOVB    #INERR,PCSR1                    ;TELL HOST THIS TEST FAILED
002222' 000462                                  BR      200$                            ;LEAVE THIS SUBTEST
002224' 000300                          30$:    SWAB    R0                              ;NEW GOOD DATA PATTERN
002226' 010011                                  MOV     R0,(R1)                         ;STORE AT SAME PLACE
002230' 011104                                  MOV     (R1),R4                         ;READ IT BACK
002232' 020400                                  CMP     R4,R0                           ;R0=GOOD DATA, R4=DATA READ
002234' 001411                                  BEQ     40$                             ;IF SAME OK
002236' 112767  000000  176657                  MOVB    #DATERR,ERRTYP                  ;INDICATE DATA ERROR
002244' 010003                                  MOV     R0,R3                           ;GET GOOD DATA
002246' 000261                                  SEC                                     ;FAILURE
002250' 112767  000003  176250                  MOVB    #INERR,PCSR1                    ;TELL HOST THIS TEST FAILED
002256' 000444                                  BR      200$                            ;LEAVE THIS SUBTEST
002260' 000300                          40$:    SWAB    R0                              ;SWITCH GOOD DATA AGAIN
002262' 001032                                  BNE     90$                             ;IF ORIGINAL PATTERN THEN WE ARE...
                                                                                        ;READING THE MEMORY TO CONTAIN A...
                                                                                        ;BACKGROUND OF LOW BYTE = ALL 1'S...
                                                                                        ;AND WRITING IT BACK SWAPPED AND...
                                                                                        ;VERIFING SWAPPED DATA
002264' 005703                          50$:    TST     R3                              ;
002266' 001032                                  BNE     100$                            ;
002270' 062701  000002                  60$:    ADD     #2,R1                           ;WE ARE GOING MIN->MAX SO ADJUST POINTER
002274' 020127  177774                          CMP     R1,#LINADR+LINSIZ               ;WE AT MAX?
002300' 103015                                  BHIS    80$                             ;YES
002302' 011104                          70$:    MOV     (R1),R4                         ;READ DATA
```

```
002304' 020004                         CMP     R0,R4                ;R0=GOOD DATA, R4=DATA READ
002306' 001746                         BEQ     30$                  ;OK IF SAME
002310' 112767  000000  176605         MOVB    #DATERR,ERRTYP       ;INDICATE DATA ERROR
002316' 010003                         MOV     R0,R3                ;GET GOOD DATA
002320' 000261                         SEC                          ;INDICATE FAILURE
002322' 112767  000003  176176         MOVB    #INERR,PCSR1         ;TELL HOST THIS TEST FAILED
002330' 000417                         BR      200$                 ;LEAVE THIS SUBTEST
002332' 000734                         BR      30$
002334' 000300                 80$:    SWAB    R0                   ;SWITCH GOOD DATA
002336' 001716                         BEQ     20$                  ;IF LOW BYTE = ALL 0'S GO IN MAX->MIN
002340' 005103                         COM     R3                   ;ELSE GO IN MIN->MAX DIRECTION
002342' 012701  100000                 MOV     #LINADR,R1           ;SET ADDRESS POINTER TO MIN
002346' 000755                         BR      70$

002350' 005703                 90$:    TST     R3                   ;ARE WE GOING MIN->MAX?
002352' 001344                         BNE     50$                  ;YES

002354' 020127  100000         100$:   CMP     R1,#LINADR           ;NO, CHECK TO SEE IF WE ARE AT MIN
002360' 101305                         BHI     20$                  ;NOT YET
002362' 000300                         SWAB    R0                   ;WE ARE AT MIN SO SWITCH GOOD DATA
002364' 001746                         BEQ     70$                  ;IF LOW BYTE = ALL 0'S ELSE WE ARE DONE
002366' 000241                         CLC                          ;INDICATE SUCCESS
002370' 012600                 200$:   MOV     (SP)+,R0             ;RESTORE R0
002372' 000207                         RTS     PC
```

```
                                        .SBTTL  MODULE C, MICROTEST #4

002374' 112767  000002  176124  MICC4:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
002402' 016737  176120  021020          MOV     PCSR1,@#IPCSR1
002410' 016737  176114  021010          MOV     IPCSR2,@#MDMA0          ;GET PCBB ADDRESS THROUGH HOST'S PCSR2
002416' 016737  176110  021012          MOV     IPCSR2+2,@#MDMA1
002424' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 HAS CONTENTS OF HOST'S PCSR2
002430' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 HAS CONTENTS OF HOST'S PCSR3
002434' 010037  021010                  MOV     R0,@#MDMA0              ;SETUP TO READ PCBB+0
002440' 010137  021012                  MOV     R1,@#MDMA1
002444' 013737  021014  021004          MOV     @#MDMAR0,@#DMATO        ;GET DATA PATTERN AND WRITE IT TO...
                                                                        ;DMA 'TO' ADDRESS REGISTER
002452' 013737  021004  021026          MOV     @#DMATO,@#MDMAW0        ;READ DATA PATTERN BACK AND WRITE...
                                                                        ;BACK INTO HOST MEMORY
002460' 112767  000004  176041          MOVB    #4,PCSR1+1              ;TELL HOST WHAT TEST WE JUST FINISHED
002466' 016737  176034  021020          MOV     PCSR1,@#IPCSR1
002474' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST WE ARE DONE
002502' 000241                          CLC
002504' 000207                          RTS     PC
```

```
                                         .SBTTL  MODULE C, MICROTEST #5


002506' 112767  000002  176012  MICC5:  MOVB    #!NTST,PCSR1            ;TELL HOST WE ARE TESTING
002514' 016737  176006  021020          MOV     PCSR1,@#IPCSR1
002522' 012700  100000                  MOV     #LINADR,R0              ;BASE ADDRESS OF LINK MEMORY
002526' 012701  037776                  MOV     #LINSIZ/2,R1            ;NUMBER OF WORDS IN LINK MEMORY
002532' 010010                  10$:    MOV     R0,(R0)                 ;FILL LINK MEMORY WITH THE ADDRESS
002534' 005720                          TST     (R0)+
002536' 005301                          DEC     R1                      ;OF EACH LOCATION
002540' 001374                          BNE     10$
002542' 016737  175762  021010          MOV     IPCSR2,@#MDMA0          ;GET PCBB THROUGH HOST'S PCSR2
002550' 016737  175756  021012          MOV     IPCSR2+2,@#MDMA1
002556' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 HAS CONTENTS OF HOST'S PCSR2
002562' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 HAS CONTENTS OF HOST'S PCSR3
002566' 010037  021010                  MOV     R0,@#MDMA0              ;R0 HAS ADDRESS OF PCBB+0
002572' 010137  021012                  MOV     R1,@#MDMA1              ;R1 HAS ITS HIGH ORDER BITS
002576' 013737  021014  021022          MOV     @#MDMAR0,@#DMAF         ;LOAD DMA 'FROM' ADDRESS REGISTER
                                                                        ;WITH ADDRESS SUPPLIED FROM HOST
002604' 062700  000002                  ADD     #2,R0                   ;LOAD DMA 'TO' ADDRESS REGISTER
002610' 005501                          ADC     R1                      ;WITH PCBB+2 ADDRESS
002612' 010037  021004                  MOV     R0,@#DMAT0
002616' 010137  021006                  MOV     R1,@#DMAT1
002622' 012737  000002  021024          MOV     #2,@#DMAWC              ;LOAD WORD COUNT TO TRANSFER 1 WORD
002630' 005067  175704                  CLR     DMDONE                  ;CLEAR DMA DONE FLAG
002634' 005237  021002                  INC     @#DMACSR                        ;START DMA ENGINE
002640' 106427  000240                  MTPS    #PRI05                  ;LOWER CPU PRIORITY
002644' 005767  175670          20$:    TST     DMDONE                  ;IS THE TRANSFER COMPLETE?
002650' 001775                          BEQ     20$                     ;NO, WAIT FOR THE INTERRUPT
002652' 106427  000340                  MTPS    #PRI07
002656' 112767  000005  175643          MOVB    #5,PCSR1+1              ;TELL HOST WHAT TEST WE JUST FINISHED
002664' 016737  175636  021020          MOV     PCSR1,@#IPCSR1
002672' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST WE ARE DONE
002700' 000241                          CLC
002702' 000207                          RTS     PC
```

```
                                        .SBTTL  MODULE C, MICROTEST #6


002704' 112767  000002  175614  MICC6:  MOVB    #!NTST,PCSR1            ;TELL HOST WE ARE TESTING
002712' 016737  175610  021020          MOV     PCSR1,@#IPCSR1
002720' 012700  100000                  MOV     #LINADR,R0              ;GET BASE ADDRESS OF LINK MEMORY
002724' 012701  037776                  MOV     #LINSIZ/2,R1            ;SIZE OF LINK MEMORY IN WORDS
002730' 010010                  10$:    MOV     R0,(R0)                 ;FILL LINK MEMORY WITH ADDRESS OF
002732' 005720                          TST     (R0)+
002734' 005301                          DEC     R1                      ;EACH LOCATION
002736' 001374                          BNE     10$
002740' 016737  175564  021010          MOV     IPCSR2,@#MDMA0          ;GET CONTENTS OF HOST'S PCSR2
002746' 016737  175560  021012          MOV     IPCSR2+2,@#MDMA1
002754' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 HAS CONTENTS OF HOST'S PCSR2
002760' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 HAS CONTENTS OF HOST'S PCSR3
002764' 010037  021010                  MOV     R0,@#MDMA0              ;SETUP TO READ CONTENTS OF PCBB+0
002770' 010137  021012                  MOV     R1,@#MDMA1
002774' 013737  021014  021004          MOV     @#MDMAR0,@#DMATO        ;SETUP DMA 'TO' REGISTER FROM PCBB+0
003002' 012737  100000  021022          MOV     #LINADR,@#DMAF          ;SETUP DMA 'FROM' WITH LINK MEM BASE
003010' 012737  003774  021024          MOV     #3774,@#DMAWC           ;SETUP MAXIMUM BYTE COUNT
003016' 005067  175516                  CLR     DMDONE
003022' 005237  021002                  INC     @#DMACSR                ;TURN ON DMA ENGINE
003026' 106427  000240                  MTPS    #PRI05                  ;ALLOW A DMA DONE INTERRUPT
003032' 005767  175502          20$:    TST     DMDONE                  ;WAIT FOR THE INTERRUPT FROM THE DMA
003036' 001775                          BEQ     20$                     ;ENGINE
003040' 112767  000006  175461          MOVB    #6,PCSR1+1              ;TELL HOST WHAT TEST WE JUST FINISHED
003046' 016737  175454  021020          MOV     PCSR1,@#IPCSR1
003054' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST THAT THIS TEST IS DONE
003062' 000241                          CLC
003064' 000207                          RTS     PC
```

```
                                          ;
                                          .SBTTL  <DMA "TO" REGISTER RIPPLE TEST>
                                          ;
                                          ;
003066' 112767  000002  175432  MICC7:: MOVB    #INIST,PCSR1
003074' 016737  175426  021020          MOV     PCSR1,@#IPCSR1
                                          ;
                                          ;
003102' 012700  100004                  MOV     #LINADR+4,R0
003106' 012720  052525                  MOV     #052525,(R0)+
003112' 012710  031463                  MOV     #031463,(R0)
                                          ;
                                          ;
003116' 016737  175406  021010          MOV     IPCSR2,@#MDMA0
003124' 016737  175402  021012          MOV     IPCSR2+2,@#MDMA1
003132' 013700  021014                  MOV     @#MDMAR0,R0
003136' 013701  021014                  MOV     @#MDMAR0,R1
                                          ;
                                          ;
003142' 010037  021004                  MOV     R0,@#DMAT0
003146' 012737  100004  021022          MOV     #LINADR+4,@#DMAF
003154' 012737  000004  021024          MOV     #4,@#DMAWC
003162' 010700                          MOV     PC,R0
003164' 062700  175222                  ADD     #DMAINT-.,R0
003170' 010037  000114                  MOV     R0,@#DMAVEC
003174' 012737  000300  000116          MOV     #PRI06,@#DMAVEC+2
003202' 005067  175332                  CLR     DMDONE
003206' 005237  021002                  INC     @#DMACSR
003212' 106427  000240                  MTPS    #PRI05
003216' 005767  175316          20$:    TST     DMDONE
003222' 001775                          BEQ     20$
                                          ;
                                          ;
003224' 112767  000007  175275          MOVB    #7,PCSR1+1
003232' 016737  175270  021020          MOV     PCSR1,@#IPCSR1
003240' 012737  004000  021000          MOV     #DNI,@#IPCSR0
003246' 000241                          CLC
003250' 000207                          RTS     PC

003252' 003254                  MICCSZ::MICCSZ-MICROC+2                 ;SIZE OF MICROCODE MODULE C

        000001                  .END
```

```
ADDREG= 177774          103#
ADRERR= 000001          101#    428     538     605
BAKPAT  000542R   002   245#    511*    514     517     543     571*
BIT0  = 000001           51#     81
BIT1  = 000002           50#     82
BIT10 = 002000           41#     53      70
BIT11 = 004000           40#     69
BIT12 = 010000           39#     53      68      89
BIT13 = 020000           38#     67      88
BIT14 = 040000           37#     66      87     212
BIT15 = 100000           36#     65      86     217     220
BIT2  = 000004           49#     83
BIT3  = 000010           48#     84
BIT4  = 000020           47#     85
BIT5  = 000040           46#
BIT6  = 000100           45#
BIT7  = 000200           44#
BIT8  = 000400           43#     53      71
BIT9  = 001000           42#
CLRC    000544R   002   230     251#
CLRFIF= 021034           22#
CMDREG= 177776          104#    125*
CSRFLG= 000001           81#    173     199
CSRVEC= 000064           74#    149*    150*
CSRWRT  000352R   002   148     199#
C3STBL  001104R   002   305     335#
DATERR= 000000          100#    378     435     473     652     662     681
DMACSR= 021002            9#    211     220*    749*    780*    816*
DMAF  = 021022           17#    741*    777*    809*
DMAINT  000406R   002   143     211#    812
DMAT0 = 021004           10#    714*    716     745*    776*    808*
DMAT1 = 021006           11#    746*
DMAVEC= 000114           75#    144*    145*    813*    814*
DMAWC = 021024           18#    747*    778*    810*
DMDONE  000540R   002   211*    212     217     244#    748*    751     779*    782     815*    818
DNI   = 004000           69#    166     283     332     720     756     786     824
ERRFLG= 000002           82#    202
ERRINT  000362R   002   130     202#
ERRTYP  001123R   002   344#    378*    428*    435*    473*    538*    605*    652*    662*    681*
FATI  = 000400           71#
FLG3    000522R   002   165*    169     173     193*    199*    202*    225*    239#    372
INERR = 000003           80#    191     377     429     436     475     541     608     655     665     684
INMON = 000001           78#    127     189
INTST = 000002           79#    256     274     302     706     727     762     793
IOADR = 020000           97#     98
IOSIZ = 020000           93#     98
IPCSR0= 021000            8#    166*    178     203*    214*    219*    226*    267*    283*    332*    720*    756*    786*
                        824*
IPCSR1= 021020           16#    128*    192*    257*    275*    282*    303*    331*    707*    719*    728*    755*    763*
                        785*    794*    823*
IPCSR2  000530R   002   163*    164*    242#    258     259     315     316     708     709     735     736     770     771
                        802     803
LASFTP= 012400           53#
LFRBUF= 021032           21#
LINADR= 100000           99#    102     361     370     383     386     388     412     413     466     513     515     521
                        523     528     530     569     593     599     615     643     646     676     690     696     729
```

```
                               764     777     797     809
LINSIZ= 077774                  95#    102     361     362     383     388     515     523     530     569     599     646     676
                               730     765
LRBUF  = 021044                 26#
LTAC   = 021030                 20#
MDMAR0= 021014                  14#    260     261     267     317     318     710     711     714     737     738     741     772
                               773     776     804     805
MDMAR1= 021016                  15#
MDMAW0= 021026                  19#    321*    323*    324*    325*    716*
MDMAW1= 021036                  23#
MDMA0  = 021010                 12#    258*    262*    315*    319*    708*    712*    735*    739*    770*    774*    802*
MDMA1  = 021012                 13#    259*    263*    316*    320*    709*    713*    736*    740*    771*    775*    803*
MDMSR  = 021042                 25#
MICCSZ  003252RG      002      828#
MICC1   000554R       002      231     256#
MICC2   000644R       002      232     274#
MICC3   000732R       002      233     302#
MICC3A  001124R       002      335     356#
MICC3B  001266R       002      336     409#
MICC3C  001432R       002      337     464#
MICC3D  001550R       002      338     509#
MICC3E  002020R       002      339     590#
MICC3F  002142R       002      340     640#
MICC4   002374R       002      234     706#
MICC5   002506R       002      235     727#
MICC6   002704R       002      236     762#
MICC7   003066RG      002      237     793#
MICROC  000000RG      002      124#    828
MODREG= 177774                 102#    103     104
NPRERR= 100000                  86#    219
NPRFLG= 000020                  85#
NXMERR= 040000                  87#    214
NXMFLG= 000010                  84#
PARERR= 010000                  89#    226
PARFLG= 000004                  83#    225     372
PARINT  000464R       002      138     225#
PARVEC= 000140                  76#    139*    140*
PCBADR  000534R       002      243#
PCEI   = 040000                 66#
PCSRSW= 021040                  24#    157
PCSR1   000526R       002      127*    128     189*    191*    192     241#    251*    256*    257     268*    274*    275     281*
                               282     302*    303     330*    331     377*    429*    436*    475*    541*    608*    655*    665*
                               684*    706*    707     718*    719     727*    728     754*    755     762*    763     784*    785
                               793*    794     822*    823
PHYAD0= 021060                  27#
PHYAD1= 021062                  28#
PHYAD2= 021064                  29#
PHYAD3= 021066                  30#
PHYAD4= 021070                  31#
PHYAD5= 021072                  32#
PRI00  = 000000                 61#    167*
PRI01  = 000040                 60#
PRI02  = 000100                 59#
PRI03  = 000140                 58#
PRI04  = 000200                 57#    150     277*
PRI05  = 000240                 56#    155     750*    781*    817*
```

```
PRI06 = 000300              55#     145     358*    814
PRI07 = 000340              54#     124*    133     140     172*    280*    753*
RCEI  = 002000              70#
ROMADR= 040000              98#      99
ROMSIZ= 040000              94#      99
RXI   = 020000              67#
SANTIM  000524R    002     208*     240#    276*    278
SAWVEC= 000134              73#     154*    155*
SERI  = 100000              65#
SIZ4K = 020000              90#      91      92      93     489     556     565
SIZ8K = 040000              91#      94      95
STACK = 001000              77#     126
SUBNUM  001122R    002     321      343#    357*    410*    465*    510*    591*    641*
TBLC    000502R    002     183      230#
TIMINT  000400R    002     153      208#
TXI   = 010000              68#
UNIERR= 020000              88#     203
WCSADR= 000000              96#      97
WCSSIZ= 020000              92#      97
.     = 003254R    002     130      138     143     148     153     175     183     205     221     230     231     232     233
                           234      235     236     237     305     335     336     337     338     339     340     368     812
```

```
BCOMPL      1#
BERROR      1#
BGNAU       1#
BGNAUT      1#
BGNCLN      1#
BGNDU       1#
BGNHRD      1#
BGNHW       1#
BGNINI      1#
BGNMOD      1#
BGNMSG      1#
BGNPRO      1#
BGNPTA      1#
BGNRPT      1#
BGNSEG      1#
BGNSET      1#
BGNSFT      1#
BGNSRV      1#
BGNSUB      1#
BGNSW       1#
BGNTST      1#
BNCOMP      1#
BNERRO      1#
BREAK       1#
BRESET      1#
CKLOOP      1#
CLOCK       1#
CLOSE       1#
CLRVEC      1#
COMMEN      1#
DELAY       1#
DESCRI      1#
DEVTYP      1#
DISPAT      1#
DISPLA      1#
DOCLN       1#
DODU        1#
DORPT       1#
ENDAU       1#
ENDAUT      1#
ENDCLN      1#
ENDCOM      1#
ENDDU       1#
ENDHRD      1#
ENDHW       1#
ENDINI      1#
ENDMOD      1#
ENDMSG      1#
ENDPRO      1#
ENDPTA      1#
ENDRPT      1#
ENDSEG      1#
ENDSET      1#
ENDSFT      1#
ENDSRV      1#
ENDSUB      1#
```

```
ENDSW      1#
ENDTST     1#
EQUALS     1#
ERRDF      1#
ERRHRD     1#
ERROR      1#
ERRSF      1#
ERRSOF     1#
ERRTBL     1#
ESCAPE     1#
EXIT       1#
FEQUAL     1#
GETBYT     1#
GETPRI     1#
GETWOR     1#
GMANIA     1#
GMANID     1#
GMANIL     1#
GPHARD     1#
GPRMA      1#
GPRMD      1#
GPRML      1#
HEADER     1#
INLOOP     1#
IOSETU     1#
IOSTAR     1#
KT11       1#
LASTAD     1#
MANUAL     1#
MEMORY     1#
MSBYTE     1#
MSCHEC     1#
MSCNTO     1#
MSCOUN     1#
MSDATA     1#
MSDECR     1#
MSDEFA     1#
MSENDE     1#
MSERRI     1#
MSESCA     1#
MSESCS     1#
MSEXCP     1#
MSEXIT     1#
MSEXSE     1#
MSEXTJ     1#
MSGEN      1#
MSGENB     1#
MSGETS     1#
MSGETT     1#
MSGNGB     1#
MSGNIN     1#
MSGNLS     1#
MSGNSU     1#
MSGNTA     1#
MSGNTE     1#
MSHAPT     1#
```

```
MSHNAP     1#
MSINCR     1#
MSIOSE     1#
MSLDRO     1#
MSMASK     1#
MSMCHI     1#
MSMCLO     1#
MSMSK1     1#
MSPOP      1#
MSPRIN     1#
MSPUSH     1#
MSPUT      1#
MSPUT1     1#
MSRADI     1#
MSRBRO     1#
MSRNRO     1#
MSSETS     1#
MSSTAR     1#
MSSVC      1#
MSTLAB     1#
MSTSTL     1#
MSWORD     1#
MSXFER     1#
OPEN       1#
POINTE     1#
PRINTB     1#
PRINTF     1#
PRINTS     1#
PRINTX     1#
READBU     1#
READEF     1#
RFLAGS     1#
SETPRI     1#
SETVEC     1#
SLASH      1#
STARS      1#
SVC        1#
XFER       1#
XFERF      1#
XFERT      1#


. ABS.  000000     000
        000000     001
MICRC   003254     002


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROC.OBJ,MICROC.LST/CR/SOL/NL:TOC=SVC34R.P11,MICROC.MAC
 RUN-TIME: 2 3 .4 SECONDS
 RUN-TIME RATIO: 39/6=5.9
 CORE USED:  31K  (61 PAGES)
```

```
                        .TITLE  MICROD - MICROCODE MODULE D

        000000'         .CSECT MICRD

                        .SBTTL  REGISTER DEFINITIONS USED BY THE T11

        021000          IPCSR0  =       21000   ;INTERNAL PCSR0 ADDRESS
        021002          DMACSR  =       21002   ;DMA ENGINE CONTROL STATUS REGISTER
        021004          DMAT0   =       21004   ;DMA ENGINE TO ADDRESS REGISTER #0
        021006          DMAT1   =       21006   ;DMA ENGINE TO ADDRESS REGISTER #1
        021010          MDMA0   =       21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
        021012          MDMA1   =       21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
        021014          MDMAR0  =       21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
        021016          MDMAR1  =       21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
        021020          IPCSR1  =       21020   ;INTERNAL PCSR1 ADDRESS
        021022          DMAF    =       21022   ;DMA ENGINE FROM ADDRESS REGISTER
        021024          DMAWC   =       21024   ;DMA ENGINE WORD COUNT REGISTER
        021026          MDMAW0  =       21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
        021030          LTAC    =       21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
        021032          LFRBUF  =       21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
        021034          CLRFIF  =       21034   ;CLEAR FIFO
        021036          MDMAW1  =       21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
        021040          PCSRSW  =       21040   ;SWITCH PACK REGISTER
        021042          MDMSR   =       21042   ;MICROCPU DMA STATUS REGISTER
        021044          LRBUF   =       21044   ;LINK RECIEVE BUFFER COMPLETED
        021060          PHYAD0  =       21060   ;PHYSICAL ADDRESS ROM BYTE 0
        021062          PHYAD1  =       21062   ;PHYSICAL ADDRESS ROM BYTE 1
        021064          PHYAD2  =       21064   ;PHYSCIAL ADDRESS ROM BYTE 2
        021066          PHYAD3  =       21066   ;PHYSICAL ADDRESS ROM BYTE 3
        021070          PHYAD4  =       21070   ;PHYSICAL ADDRESS ROM BYTE 4
        021072          PHYAD5  =       21072   ;PHYSICAL ADDRESS ROM BYTE 5
        177774          MODREG  =       177774  ;LINK MODE REGISTER
        177774          ADRREG  =       177774  ;LINK STATION ADDRESS RAM REGISTER
        177776          CMDREG  =       177776  ;LINK COMMAND REGISTER

                        .SBTTL  OTHER DEFINITIONS USED BY THE MICROCODE

        100000          BIT15   =       100000
        040000          BIT14   =       40000
        020000          BIT13   =       20000
        010000          BIT12   =       10000
        004000          BIT11   =       4000
        002000          BIT10   =       2000
        001000          BIT9    =       1000
        000400          BIT8    =       400
        000200          BIT7    =       200
        000100          BIT6    =       100
        000040          BIT5    =       40
        000020          BIT4    =       20
        000010          BIT3    =       10
        000004          BIT2    =       4
        000002          BIT1    =       2
        000001          BIT0    =       1
                        ;
        012400          LASFTP  =       BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
```

```
        000340          PRIO7   =       340
        000300          PRIO6   =       300
        000240          PRIO5   =       240
        000200          PRIO4   =       200
        000140          PRIO3   =       140
        000100          PRIO2   =       100
        000040          PRIO1   =       40
        000000          PRIO0   =       0
                        ;
                        ;PCSR0 - PORT CONTROL STATUS REGISTER 0
                        ;
        100000          SERI    =       BIT15
        040000          PCEI    =       BIT14
        020000          RXI     =       BIT13
        010000          TXI     =       BIT12
        004000          DNI     =       BIT11
        002000          RCEI    =       BIT10
        000400          FATI    =       BIT8
                        ;
                        ;LINK COMMAND REGISTER
                        ;
        100000          ENABLE  =       BIT15   ;ENABLE LINK MODULE
        000200          MODE    =       BIT7    ;ENABLE MODE REGISTER
        000100          ARAM    =       BIT6    ;ENABLE STATION ADDRESS RAM
                        ;
                        ;LINK MODE REGISTER
                        ;
        100000          PROM    =       BIT15   ;PROMISCUIOUS MODE
        040000          ENAL    =       BIT14   ;ENABLE MULTICAST
        004000          ENCR    =       BIT11   ;ENABLE COLLISION TEST
        002000          ACLO    =       BIT10   ;ENABLE ACLO
        000040          DRTY    =       BIT5    ;DISABLE RETRY LOGIC
        000020          COLL    =       BIT4    ;SIMULATE A COLLISION
        000010          DTCR    =       BIT3    ;DISABLE TRANSMIT CRC LOGIC
        000004          LOOP    =       BIT2    ;ENABLE LOOPBACK

        000070          TRNVEC= 70              ;VECTOR ADDRESS FOR THE TRANSMITTER
        000120          RCVVEC= 120             ;VECTOR ADDRESS FOR THE RECEIVER
        000134          SANVEC= 134             ;VECTOR ADDRESS FOR THE SANITY TIMER
        000064          CSRVEC= 64              ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
        000114          DMAVEC= 114             ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
        000140          PARVEC= 140             ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
        001000          STACK=  1000            ;STACK LOCATION
        000001          INMON=  1               ;IN MICROMONITOR STATE
        000002          INTST=  2               ;IN A TEST STATE
        000003          INERR=  3               ;IN ERROR STATE
        000001          CSRFLG= BIT0            ;CSR WRITE INTERRUPT OCCURED
        000002          ERRFLG= BIT1            ;UNEXPECTED ERROR OCCURED
        000004          PARFLG= BIT2            ;PARITY ERROR OCCURED
        000010          NXMFLG= BIT3            ;NON-EXISTANT MEMORY ERROR OCCURRED
        000020          NPRFLG= BIT4            ;NPR TIMEOUT OCCURRED
        000040          TRNFLG= BIT5            ;TRANSMITTER INTERRUPT OCCURRED
        000100          RCVFLG= BIT6            ;RECEIVER INTERRUPT OCCURRED
        100000          NPRERR= BIT15           ;PCSR0 FLAG INDICATING NPR ERROR ERROR OCCURRED
        040000          NXMERR= BIT14           ;PCSR0 FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
        020000          UNIERR= BIT13           ;PCSR0 FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
```

OTHER DEFINITIONS USED BY THE MICROCODE

```
        010000          PARERR= BIT12           ;PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
        004000          SIZ1K=  4000            ;1K WORDS
        010000          SIZ2K=  SIZ1K*2         ;2K WORDS
        014000          SIZ3K=  SIZ1K*3         ;3K WORDS
        020000          SIZ4K=  SIZ2K*2         ;4K WORDS
        040000          SIZ8K=  SIZ4K*2         ;8K WORDS
        020000          WCSSIZ= SIZ4K           ;SIZE OF WRITEABLE CONTROL STORE
        020000          IOSIZ=  SIZ4K           ;SIZE OF I/O PAGE
        040000          ROMSIZ= SIZ8K           ;SIZE OF ROM
        077774          LINSIZ= SIZ8K*2-4       ;SIZE OF LINK MEMORY
        000000          WCSADR= 0               ;BASE ADDRESS OF WCS
        020000          IOADR=  WCSADR+WCSSIZ   ;BASE ADDRESS OF I/O PAGE
        040000          ROMADR= IOADR+IOSIZ     ;BASE ADDRESS OF ROM
        100000          LINADR= ROMADR+ROMSIZ   ;BASE ADDRESS OF LINK MEMORY
        002756          MAXBC=  1518.           ;MAXIMUM # OF BYTES IN A TRANSMIT BUFFER
        007777          MXMTBC= 7777            ; MAXIMUM COUNT IN LINK XMIT BYTE COUNT FIELD
        007777          MRECBC= 7777            ; MAXIMUM COUNT IN LINK RECEIVE "MLEN" FIELD
        000004          CRCSIZ= 4               ;NUMBER OF BYTES IN THE CRC
        000000          DATERR= 0               ;FLAG INDICATING DATA ERROR OCCURRED
        000001          ADRERR= 1               ;FLAG INDICATING ADDRESS ERROR OCCURRED
```

```
                                       .SBTTL  D_MODULE MICROCODE

000000' 106427  000340                 MICROD::MTPS    #PRI07                  ;DISABLE INTERRUPTS
000004' 012737  000000  177776                 MOV     #0,@#CMDREG             ;TURN OFF THE LINK
000012' 012706  001000                         MOV     #STACK,SP               ;SETUP STACK
000016' 112767  000001  000620                 MOVB    #INMON,PCSR1            ;TELL HOST WE ARE IN MICROMONITOR
000024' 016737  000614  021020                 MOV     PCSR1,@#IPCSR1
000032' 012737  004000  021000                 MOV     #DNI,@#IPCSR0           ;TELL HOST THE LOAD AND START FINISHED
000040' 010700                                 MOV     PC,R0                   ;GET ADDRESS OF UNEXPECTED ERROR...
000042' 062700  000404                         ADD     #ERRINT-.,R0            ;HANDLER
000046' 005001                                 CLR     R1                      ;FILL ALL UNUSED VECTORS WITH TRAP...
000050' 010021                         10$:    MOV     R0,(R1)+                ;HANDLER
000052' 012721  000340                         MOV     #PRI07,(R1)+
000056' 020127  001000                         CMP     R1,#1000
000062' 002772                                 BLT     10$

000064' 010700                                 MOV     PC,R0                   ;SETUP PARITY TRAP VECTOR
000066' 062700  000462                         ADD     #PARINT-.,R0
000072' 010037  000140                         MOV     R0,@#PARVEC
000076' 012737  000340  000142                 MOV     #PRI07,@#PARVEC+2

000104' 010700                                 MOV     PC,R0                   ;SETUP DMA INTERRUPT VECTOR
000106' 062700  000364                         ADD     #DMAINT-.,R0
000112' 010037  000114                         MOV     R0,@#DMAVEC
000116' 012737  000340  000116                 MOV     #PRI07,@#DMAVEC+2

000124' 010700                                 MOV     PC,R0                   ;SETUP CSR WRITE VECTOR
000126' 062700  000310                         ADD     #CSRWRT-.,R0
000132' 010037  000064                         MOV     R0,@#CSRVEC
000136' 012737  000200  000066                 MOV     #PRI04,@#CSRVEC+2

000144' 010700                                 MOV     PC,R0                   ;SETUP SANTITY TIMER VECTOR
000146' 062700  000316                         ADD     #TIMINT-.,R0
000152' 010037  000134                         MOV     R0,@#SANVEC
000156' 012737  000240  000136                 MOV     #PRI05,@#SANVEC+2

000164' 010700                                 MOV     PC,R0                   ;SETUP TRANSMITTER VECTOR
000166' 062700  000414                         ADD     #TRNINT-.,R0
000172' 010037  000070                         MOV     R0,@#TRNVEC
000176' 012737  000200  000072                 MOV     #PRI04,@#TRNVEC+2

000204' 010700                                 MOV     PC,R0                   ;SETUP RECEIVER VECTOR
000206' 062700  000360                         ADD     #RCVINT-.,R0
000212' 010037  000120                         MOV     R0,@#RCVVEC
000216' 012737  000240  000122                 MOV     #PRI05,@#RCVVEC+2

000224' 013700  021040                         MOV     @#PCSRSW,R0             ;GET SWITCH PACK BITS
000230' 052700  176000                         BIS     #176000,R0              ;MAP THEM INTO HOST I/O PAGE
000234' 006300                                 ASL     R0                      ;SHIFT OVER TO POSITION CORRECTLY
000236' 006300                                 ASL     R0
000240' 006300                                 ASL     R0
000242' 062700  000004                         ADD     #4,R0                   ;PCSR2 IS PCSR0+4
000246' 010067  000374                         MOV     R0,IPCSR2               ;SAVE PCSR2 ADDRESS
000252' 012767  000003  000370                 MOV     #3,IPCSR2+2             ;HIGH ORDER BITS 17:16
000260' 005067  000354                         CLR     FLG4                    ;INITIALIZE FLAG WORD
```

```
000264' 106427 000000          15$:    MTPS    #PRI00                  ;ALLOW INTERRUPTS
000270' 005767 000344          20$:    TST     FLG4                    ;WAIT FOR A COMMAND FROM HOST
000274' 001775                         BEQ     20$
000276' 106427 000340                  MTPS    #PRI07                  ;RAISE CPU PRIORITY TO SERVICE COMMAND
000302' 032767 000001 000330           BIT     #CSRFLG,FLG4            ;DID HOST GIVE US A COMMAND?
000310' 001001                         BNE     30$                     ;YES
000312' 000777                         BR      .                       ;NO, ERROR SO JUST SIT HERE...
                                                                       ;FOR LACK OF ANYTHING BETTER TO DO
000314' 113700 021000          30$:    MOVB    @#IPCSR0,R0             ;GET WHAT HOST WROTE TO PCSR0
000320' 042700 177760                  BIC     #177760,R0              ;STRIP ALL BUT COMMAND BITS
000324' 001004                         BNE     35$                     ;WAS IT THE CLEAR FUNCTION?
000326' 012737 000001 021020           MOV     #INMON,@#IPCSR1         ;YES, CLEAR OUT THE TEST # BITS
000334' 000432                         BR      50$
000336' 022700 000017          35$:    CMP     #17,R0                  ;START OPERATIONAL MICROCODE?
000342' 001432                         BEQ     60$
000344' 162700 000001                  SUB     #1,R0
000350' 010701                         MOV     PC,R1                   ;GET ADDRESS OF OUR COMMAND TABLE
000352' 062701 000240                  ADD     #TBLD-.,R1
000356' 006300                         ASL     R0                      ;MAKE COMMAND A BYTE OFFSET
000360' 060001                         ADD     R0,R1                   ;USE IT TO INDEX INTO COMMAND TABLE
000362' 061101                         ADD     (R1),R1                 ;R1 NOW HAS COMMAND ROUTINE ADDRESS
000364' 004711                         JSR     PC,(R1)                 ;EXECUTE AS COMMANDED FROM HOST
000366' 103404                         BCS     40$                     ;ERROR OCCURRED
000370' 112767 000001 000246           MOVB    #INMON,PCSR1            ;INDICATE TO HOST WE ARE BACK IN...
000376' 000403                         BR      45$                     ;MICROMONITR
000400' 112767 000003 000236   40$:    MOVB    #INERR,PCSR1            ;INDICATE TO HOST ERROR OCCURRED
000406' 016737 000232 021020   45$:    MOV     PCSR1,@#IPCSR1
000414' 012737 004000 021000           MOV     #DNI,@#IPCSR0           ;TELL HOST THIS MICROTEST FINISHED
000422' 005067 000212          50$:    CLR     FLG4                    ;RESET FLAG WORD
000426' 000716                         BR      15$                     ;GO WAIT FOR ANOTHER COMMAND
000430' 005000                 60$:    CLR     R0                      ;FAKE SELF TEST RESULTS
000432' 000137 040006                  JMP     @#40006                 ;START OPERATIONAL MICROCODE
000436' 052767 000001 000174   CSRWRT: BIS     #CSRFLG,FLG4            ;INDICATE A CSR WRITE INTERRUPT OCCURED
000444' 000002                         RTI
000446' 052767 000002 000164   ERRINT: BIS     #ERRFLG,FLG4            ;INDICATE A UNEXPECTED INTERRUPT OCCURED
000454' 012737 020000 021000           MOV     #UNIERR,@#IPCSR0        ;TELL HOST AN UNEXPECTED INTERRUPT
                                                                       ;HAPPENED
000462' 000777                         BR      .                       ;JUST SIT HERE AND SPIN WHEELS
                                                                       ;COUNT ON HOST TO TIME OUT
000464' 005267 000152          TIMINT: INC     SANTIM                  ;COUNT TICKS AS THEY OCCUR
000470' 000002                         RTI
000472' 013767 021002 000156   DMAINT: MOV     @#DMACSR,DMDONE         ;GET DMA STATUS
000500' 032767 040000 000150           BIT     #BIT14,DMDONE           ;DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
000506' 001404                         BEQ     10$                     ;NO
000510' 012737 040000 021000           MOV     #NXMERR,@#IPCSR0        ;YES, TELL HOST A NON-EXISTANT MEMORY
                                                                       ;LOCATION WAS ADDRESSED
000516' 000407                         BR      20$
000520' 032767 100000 000130   10$:    BIT     #BIT15,DMDONE           ;DID A NPR TIMEOUT OCCUR?
```

```
000526' 001407                          BEQ     30$                     ;NO
000530' 012737  100000  021000          MOV     #NPRERR,@#IPCSR0        ;TELL HOST NPR TIMEOUT HAPPENED
000536' 012737  100000  021002  20$:    MOV     #BIT15,@#DMACSR         ;CLEAR THE INTERRUPT IN THE DMA ENGINE
000544' 000777                          BR      .                       ;SIT HERE AND SPIN WHEELS
000546' 000002                  30$:    RTI


000550' 052767  000004  000062  PARINT: BIS     #PARFLG,FLG4            ;SET PARITY ERROR OCCURRED
000556' 012737  010000  021000          MOV     #PARERR,@#IPCSR0        ;TELL HOST A LINK MEMORY PARITY ERROR
                                                                        ;OCCURRED
000564' 000777                          BR      .                       ;SIT HERE AND SPIN WHEELS

000566' 005737  021044          RCVINT: TST     @#LRBUF                 ;READ BUFFER DONE REGISTER...
                                                                        ;WHICH CLEARS THE INTERRUPT
000572' 052767  000100  000040          BIS     #RCVFLG,FLG4            ;SET RECEIVER INTERRUPT OCCURRED
000600' 000002                          RTI

000602' 052767  000040  000030  TRNINT: BIS     #TRNFLG,FLG4            ;SET TRANSMITTER INTERRUPT OCCURRED
000610' 000002                          RTI

000612' 000056                  TBLD:   .WORD   MICD1-.                 ;TRANSMITTER DONE TEST
000614' 000236                          .WORD   MICD2-.                 ;RECEIVER DONE TESTS
000616' 000432                          .WORD   MICD3-.                 ; DATA BYTE FRAMING TEST
000620' 000734                          .WORD   MICD4-.                 ; DATA WORD FRAMING TEST
000622' 001244                          .WORD   MICD5-.                 ; DATA PATH PATTERN
000624' 001604                          .WORD   MICD6-.                 ; STATUS MUX TEST
000626' 002046                          .WORD   MICD7-.                 ; LINK BYTE COUNT TEST
000630' 002354                          .WORD   MICD8-.                 ;LINK MEMORY ARBITRATION TEST
000632' 003014                          .WORD   MICD9-.                 ;LINK BYTE COUNTER MAXIMUM TEST
000634' 003206                          .WORD   MICD10-.                ;FIFO TEST
000636' 003460                          .WORD   MICD11-.                ;LINK MEMORY ADDRESS TEST

000640' 000000                  FLG4:   .WORD   0                       ;FLAG WORD
000642' 000000                  SANTIM: .WORD   0                       ;COUNT FOR SANITY TIMER
000644' 000000                  PCSR1:  .WORD   0                       ;COPY OF WHAT GOES TO PCSR1
000646' 000000  000000          IPCSR2: .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCSR2
000652' 000000  000000          PCBADR: .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCB
000656' 000000                  DMDONE: .WORD   0
000660' 000000                  RBUFP:  .WORD   0                       ;POINTER TO RECIEVE BUFFER
000662' 000000                  TBUFP:  .WORD   0                       ;POINTER TO XMIT BUFFER
000664' 000000                  DBUFP:  .WORD   0                       ;POINTER TO DMA ENGINE BUFFER
000666' 000000                  MBUFP:  .WORD   0                       ;POINTER TO MICROCPU DMA BUFFER
```

```
000670' 112767 000002 177746  MICD1:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
000676' 016737 177742 021020          MOV     PCSR1,@#IPCSR1
000704' 012703 177777                 MOV     #177777,R3              ;FILL RECIEVE BUFFER WITH 1'S
000710' 012700 100000                 MOV     #LINADR,R0              ;RECIEVE BUFFER STARTS HERE
000714' 010067 177740                 MOV     R0,RBUFP
000720' 010320                10$:    MOV     R3,(R0)+
000722' 020027 104000                 CMP     R0,#LINADR+SIZ1K        ;FILL ENTIRE BUFFER
000726' 103774                        BLO     10$
000730' 005003                        CLR     R3                      ;FILL XMIT BUFFER WITH 0'S
000732' 010067 177724                 MOV     R0,TBUFP                ;XMIT BUFFER STARTS 1K AWAY FROM RECIEVE
000736' 010320                20$:    MOV     R3,(R0)+
000740' 020027 110000                 CMP     R0,#LINADR+SIZ2K
000744' 103774                        BLO     20$

000746' 012737 100200 177776          MOV     #MODE!ENABLE,@#CMDREG   ;ENABLE LINK MODULE AND SELECT MODE REG
000754' 012737 100004 177774          MOV     #PROM!LOOP,@#MODREG     ;SET PROMISCUIOUS MODE AND ENABLE LOOPBACK

000762' 016701 177674                 MOV     TBUFP,R1                ;POINT TO XMIT BUFFER
000766' 005021                        CLR     (R1)+                   ;CLEAR OUT STATUS WORD
000770' 012721 002752                 MOV     #MAXBC-CRCSIZ,(R1)+     ;SET BYTE COUNT TO MAXIMUM ALLOWED
000774' 005037 021034                 CLR     @#CLRFIF                ;CLEAR THE FIFO
001000' 005067 177634                 CLR     FLG4                    ;CLEAR THE INTERRUPT FLAG
001004' 016737 177650 021032          MOV     RBUFP,@#LFRBUF          ;TELL LINK WHERE RECIEVE BUFF IS
001012' 016737 177644 021030          MOV     TBUFP,@#LTAC            ;TELL LINK WHERE TRANSMIT BUFF IS...
                                                                      ;WHICH WILL START A XMIT OPERATION
001020' 106427 000140                 MTPS    #PRI03                  ;ALLOW XMITTER TO INTERRUPT
001024' 032767 000040 177606  30$:    BIT     #TRNFLG,FLG4            ;WAIT FOR XMIT INTERRUPT
001032' 001774                        BEQ     30$
001034' 106427 000340                 MTPS    #PRI07                  ;DISABLE INTERRUPTS
001040' 112767 000001 177577          MOVB    #1,PCSR1+1              ;TELL HOST THIS IS TEST #1
001046' 000241                        CLC
001050' 000207                        RTS     PC
```

```
001052' 112767  000002  177564  MICD2:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
001060' 016737  177560  021020          MOV     PCSR1,@#IPCSR1
001066' 012703  177777                  MOV     #177777,R3              ;FILL RECIEVE BUFFER WITH 1'S
001072' 012700  100000                  MOV     #LINADR,R0              ;RECIEVE BUFFER STARTS HERE
001076' 010067  177556                  MOV     R0,RBUFP
001102' 010320                  10$:    MOV     R3,(R0)+
001104' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ;FILL ENTIRE BUFFER
001110' 103774                          BLO     10$
001112' 005003                          CLR     R3                      ;FILL XMIT BUFFER WITH 0'S
001114' 010067  177542                  MOV     R0,TBUFP                ;XMIT BUFFER STARTS 1K AWAY FROM RECIEVE
001120' 010320                  20$:    MOV     R3,(R0)+
001122' 020027  110000                  CMP     R0,#LINADR+SIZ2K
001126' 103774                          BLO     20$

001130' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ;ENABLE LINK MODULE AND SELECT MODE REG
001136' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG     ;SET PROMISCUIOUS MODE AND ENABLE LOOPBACK

001144' 016701  177512                  MOV     TBUFP,R1                ;POINT TO XMIT BUFFER
001150' 005021                          CLR     (R1)+                   ;CLEAR OUT STATUS WORD
001152' 012721  002752                  MOV     #MAXBC-CRCSIZ,(R1)+     ;SET BYTE COUNT TO MAXIMUM ALLOWED
001156' 005037  021034                  CLR     @#CLRFIF                ;CLEAR THE FIFO
001162' 005067  177452                  CLR     FLG4                    ;CLEAR THE INTERRUPT FLAG
001166' 016737  177466  021032          MOV     RBUFP,@#LFRBUF          ;TELL LINK WHERE RECIEVE BUFF IS
001174' 016737  177462  021030          MOV     TBUFP,@#LTAC            ;TELL LINK WHERE TRANSMIT BUFF IS...
                                                                        ;WHICH WILL START A XMIT OPERATION
001202' 106427  000200                  MTPS    #PRI04                  ;ALLOW RECEIVER TO INTERRUPT
001206' 032767  000100  177424  30$:    BIT     #RCVFLG,FLG4            ;WAIT FOR RECEIVE INTERRUPT
001214' 001774                          BEQ     30$

001216' 106427  000140                  MTPS    #PRI03                  ; ALLOW XMTR TO INTERRUPT
001222' 032767  000040  177410  40$:    BIT     #TRNFLG,FLG4            ; WAIT FOR XMIT INTERRUPT
001230' 001774                          BEQ     40$

001232' 106427  000340                  MTPS    #PRI07                  ;DISABLE INTERRUPTS
001236' 112767  000002  177401          MOVB    #2,PCSR1+1              ;TELL HOST THIS IS TEST #2
001244' 000241                          CLC
001246' 000207                          RTS     PC
```

```
                                      ;
                                      ; DATA BYTE FRAMING TEST
                                      ;
                                      ; THIS IS "MICROCODE" FOR DATA BYTE FRAMING TEST. FILLS XMIT BUFFER
                                      ; WITH PATTERN 0000000011111111 (BINARY) AND TRANSMITS OVER LOOPBACK.
                                      ; CHECKS RECEIVE BUFFER FOR SAME PATTERN. REPORTS ERRENT PATTERN,
                                      ; OFFSET FROM FRONT OF BUFFER, AND RECEIVE BUFFER STATUS WORD TO
                                      ; HOST
                                      ;
                                      .SBTTL  MODULE D, MICROTEST #3
                                      ;
001250' 112767  000002  177366  MICD3:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
001256' 016737  177362  021020          MOV     PCSR1,@#IPCSR1

                                      ; ****************************************************
                                      ; ***** FILL THE RECEIVE BUFFER WITH BACKGROUND *****
                                      ; ****************************************************

001264' 012700  100000                  MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
001270' 010067  177364                  MOV     R0,RBUFP
001274' 005020                  10$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
001276' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ; FILL ENTIRE BUFFER
001302' 103774                          BLO     10$


                                      ; ****************************************************
                                      ; ***** FILL TRANSMIT BUFFER WITH TEST PATTERN *****
                                      ; ****************************************************

001304' 012703  000377                  MOV     #0377,R3                ; WORST CASE FOR CLOCKING
001310' 010067  177346                  MOV     R0,TBUFP
001314' 010320                  20$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
001316' 020027  110000                  CMP     R0,#LINADR+SIZ2K                ; STOP AT TOP
001322' 103774                          BLO     20$


                                      ; ********************************************************
                                      ; ***** SET UP LINK FOR DATAGRAM LOOPBACK OPERATION *****
                                      ; ********************************************************

001324' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ; ENABLE LINK, SELECT MODE REG
001332' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG     ; PROM MODE AND LOOPBACK

001340' 016701  177316                  MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
001344' 005021                          CLR     (R1)+                   ; CLEAR OUT STATUS WORD
001346' 012721  002752                  MOV     #MAXBC-CRCSIZ,(R1)+     ; SET BYTE COUNT TO MAX ALLOWED
001352' 005037  021034                  CLR     @#CLRFIF                ; CLEAR THE FIFO
001356' 005067  177256                  CLR     FLG4                    ; CLEAR INTERRUPT FLAG
001362' 016737  177272  021032          MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUFF IS
001370' 016737  177266  021030          MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUFF IS

001376' 106427  000140                  MTPS    #PRI03                  ; ALLOW XMITTER AND RECEIVER TO INTERRUPT
001402' 032767  000100  177230  30$:    BIT     #RCVFLG,FLG4            ; WAIT FOR RECEIVER INTERRUPT
001410' 001774                          BEQ     30$

001412' 032767  000040  177220  35$:    BIT     #TRNFLG,FLG4            ;WAIT FOR TRANSMIT INTERRUPT TOO
```

```
001420' 001774                          BEQ     35$

001422' 106427  000340                  MTPS    #PR107                  ; DISABLE INTERRUPTS


                                ; ***********************************************
                                ; ***** VERIFY PATTERN IN RECEIVE BUFFER *****
                                ; ***********************************************

001426' 012700  100000                  MOV     #LINADR,R0              ; VERIFY RECEIVE BUFFER CONTENT
001432' 012005                          MOV     (R0)+,R5                ; SAVE STATUS IN CASE ERROR
001434' 005004                          CLR     R4                      ; TRACK OFFSET IN CASE ERROR
001436' 062700  000002                  ADD     #2,R0                   ; DON'T NEED "LENGTH" IN BUFFER

001442' 011001                  40$:    MOV     (R0),R1                 ; READ THE BUFFER
001444' 020103                          CMP     R1,R3                   ; R3 CONTAINS ORIGINAL PATTERN
001446' 001012                          BNE     70$                     ; GO TO ERROR EXIT
001450' 005200                          INC     R0                      ; COULDN'T BUMP TILL AFTER TEST
001452' 005204                          INC     R4                      ; TRACK OFFSET FROM BUFFER START
001454' 022704  002752                  CMP     #MAXBC-CRCSIZ,R4        ; COMPARE ALL BUFFER ENTRIES
001460' 001370                          BNE     40$
001462' 000241                          CLC                             ;TELL MICROMONITOR SUCCESS

                                ; ***********************************************
                                ; ***** EXIT POINT FOR NO ERROR CONDITION *****
                                ; ***********************************************

001464' 112767  000003  177153  50$:    MOVB    #3,PCSR1+1              ; TELL HOST TEST 3 DONE
001472' 000207                          RTS     PC                      ; RETURN TO SENDER


                                ; ********************************************
                                ; ***** EXIT PO.NT FOR ERROR CONDITION *****
                                ; ********************************************

001474' 016737  177146  021010  70$:    MOV     IPCSR2,@#MDMA0          ; PICK UP ADDRESS OF PCBB
001502' 016737  177142  021012          MOV     IPCSR2+2,@#MDMA1
001510' 013700  021014                  MOV     @#MDMAR0,R0             ; R0=CONTENTS OF PCSR2
001514' 013702  021014                  MOV     @#MDMAR0,R2             ; R2=CONTENTS OF PCSR3
001520' 010037  021010                  MOV     R0,@#MDMA0              ; POINT TO PCBB+0
001524' 010237  021012                  MOV     R2,@#MDMA1

001530' 010537  021026                  MOV     R5,@#MDMAW0             ; WRITE STATUS WORD TO HOST
001534' 010337  021026                  MOV     R3,@#MDMAW0             ; WRITE ORIGNAL PATTERN BACK
001540' 010137  021026                  MOV     R1,@#MDMAW0             ; WRITE ERRENT PATTERN TO HOST
001544' 010437  021026                  MOV     R4,@#MDMAW0             ; WRITE ERROR OFFSET TO HOST

001550' 000261                          SEC                             ;TELL MICROMONITOR ERROR OCCURRED
001552' 000744                          BR      50$                     ; GO EXIT THROUGH NORMAL
```

```
                                        ; DATA WORD FRAMING TEST
                                        ;
                                        ; THIS IS "MICROCODE" FOR DATA WORD FRAMING TEST. FILLS XMIT BUFFER
                                        ; WITH PATTERN 0000u000000000001111111111111111 (BINARY) AND TRANSMITS
                                        ; OVER THE LOOPBACK.
                                        ;
                                        ; CHECKS RECEIVE BUFFER FOR SAME PATTERN. REPORTS ERRENT PATTERN,
                                        ; OFFSET FROM FRONT OF BUFFER, AND RECEIVE BUFFER STATUS WORD TO
                                        ; HOST
                                        ;
                                        .SBTTL  MODULE D,MICROTEST #4

                                        ; **********************************
                                        ; ***** TELL HOST WE ARE BUSY *****
                                        ; **********************************

001554' 112767  000002  177062  MICD4:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
001562' 016737  177056  021020          MOV     PCSR1,@#IPCSR1


                                        ; *****************************************************
                                        ; ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
                                        ; *****************************************************

001570' 012700  100000                  MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
001574' 010067  177060                  MOV     R0,RBUFP
001600' 005020                  10$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
001602' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ; FILL ENTIRE BUFFER
001606' 103774                          BLO     10$


                                        ; ******************************************
                                        ; ***** FILL XMIT BUFFER WITH TEST PATTERN *****
                                        ; ******************************************

001610' 012703  177777                  MOV     #177777,R3              ; WORST CASE FOR CLOCKING
001614' 010067  177042                  MOV     R0,TBUFP                ; SAVE COPY OF ADDRESS
001620' 010320                  20$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
001622' 005103                          COM     R3                      ; FLIP IT OVER
001624' 020027  110000                  CMP     R0,#LINADR+SIZ2K                ; STOP AT TOP
001630' 103773                          BLO     20$


                                        ; ******************************************
                                        ; ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
                                        ; ******************************************

001632' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ; ENABLE LINK, SELECT MODE REG
001640' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG     ; PROM MODE AND LOOPBACK

001646' 016701  177010                  MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
001652' 005021                          CLR     (R1)+                   ; CLEAR OUT STATUS WORD
001654' 012721  002752                  MOV     #MAXBC-CRCSIZ,(R1)+     ; SET BYTE COUNT TO MAX ALLOWED
001660' 005037  021034                  CLR     @#CLRFIF                ; CLEAR THE FIFO
001664' 005067  176750                  CLR     FLG4                    ; CLEAR INTERRUPT FLAG
```

```
001670' 016737  176764  021032           MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUFF IS
001676' 016737  176760  021030           MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUFF IS

001704' 106427  000140                   MTPS    #PRI03                  ; ALLOW XMITTER AND RECEIVER TO INTERRUPT
001710' 032767  000100  176722  30$:     BIT     #RCVFLG,FLG4            ; WAIT FOR RECEIVER INTERRUPT
001716' 001774                           BEQ     30$

001720' 032767  000040  176712  35$:     BIT     #TRNFLG,FLG4            ;WAIT FOR XMIT INTERRUPT TOO
001726' 001774                           BEQ     35$

001730' 106427  000340                   MTPS    #PRI07                  ; DISABLE INTERRUPTS


                                 ; ****************************************************
                                 ; ***** VERIFY THE CONTENTS OF RECEIVE BUFFER *****
                                 ; ****************************************************

001734' 012700  100000                   MOV     #LINADR,R0              ; VERIFY RECEIVE BUFFER CONTENT
001740' 012005                           MOV     (R0)+,R5                ; SAVE STATUS IN CASE ERROR
001742' 005004                           CLR     R4                      ; TRACK OFFSET IN CASE ERROR
001744' 062700  000002                   ADD     #2,R0                   ; DON'T NEED ''LENGTH'' IN BUFFER

001750' 005003                           CLR     R3                      ; NEED A ZERO
001752' 005103                  40$:     COM     R3                      ; FLIP IT OVER
001754' 011001                           MOV     (R0),R1                 ; READ DATA BACK
001756' 020103                           CMP     R1,R3
001760' 001012                           BNE     70$                     ; GO TO ERROR EXIT
001762' 005200                           INC     R0                      ; NEW ADDRESS AFTER TESTING
001764' 005204                           INC     R4                      ; NEXT OFFSET
001766' 022704  002752                   CMP     #MAXBC-CRCSIZ,R4
001772' 103767                           BLO     40$                     ; END OF LOOP
001774' 000241                           CLC                             ;TELL MICROMONITOR SUCCESS

                                 ; ****************************************
                                 ; ***** FALLTHROUGH EXIT IF NO ERROR *****
                                 ; ****************************************

001776' 112767  000004  176641  50$:     MOVB    #4,PCSR1+1              ; TELL HIM TEST FINISHED
002004' 000207                           RTS     PC                      ; RETURN TO SENDER

                                 ; **********************
                                 ; ***** ERROR EXIT *****
                                 ; **********************

002006' 016737  176634  021010  70$:     MOV     IPCSR2,@#MDMA0          ; PICK UP ADDRESS OF PCBB
002014' 016737  176630  021012           MOV     IPCSR2+2,@#MDMA1
002022' 013700  021014                   MOV     @#MDMAR0,R0             ; R0=CONTENTS OF PCSR2
002026' 013702  021014                   MOV     @#MDMAR0,R2             ; R2=CONTENTS OF PCSR3
002032' 010037  021010                   MOV     R0,@#MDMA0              ; POINT TO PCBB+0
002036' 010237  021012                   MOV     R2,@#MDMA1

002042' 010537  021026                   MOV     R5,@#MDMAW0             ; WRITE STATUS WORD TO HOST
002046' 010337  021026                   MOV     R3,@#MDMAW0             ; WRITE ORIGNAL TO HOST
002052' 010137  021026                   MOV     R1,@#MDMAW0             ; WRITE ERRENT PATTERN TO HOST
002056' 010437  021026                   MOV     R4,@#MDMAW0             ; WRITE ERROR OFFSET TO HOST
```

```
577  002062' 000261          SEC                 ;TELL MICROMONITOR ERROR OCCURRED
578  002064' 000744          BR      50$         ; GO EXIT THROUGH NORMAL
579
```

```

                                        ; DATA PATH PATTERN TEST
                                        ;
                                        ; THIS IS "MICROCODE" FOR DATA PATH PATTERN TEST. RETRIEVES DATA PATTERN
                                        ; FROM HOST MEMORY. FILLS XMIT BUFFER WITH PATTERN AND SENDS DATAGRAM
                                        ; OVER THE LOOPBACK.
                                        ;
                                        ; CHECKS RECEIVE BUFFER FOR SAME PATTERN. REPORTS ERRENT PATTERN,
                                        ; OFFSET FROM FRONT OF BUFFER, AND RECEIVE BUFFER STATUS WORD TO
                                        ; HOST

                                        .SBTTL  MODULE D,MICROTEST #5

                                        ; **********************************
                                        ; ***** TELL HOST WE ARE BUSY *****
                                        ; **********************************

 002066' 112767  000002  176550  MICD5:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
 002074' 016737  176544  021020          MOV     PCSR1,@#IPCSR1

                                        ; ********************************************
                                        ; ***** RETRIEVE PATTERN FROM HOST MEMORY *****
                                        ; ********************************************

 002102' 016737  176540  021010          MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB ADDRESS
 002110' 016737  176534  021012          MOV     IPCSR2+2,@#MDMA1
 002116' 013700  021014                  MOV     @#MDMAR0,R0             ; R0 NOW CONTAINS PCBB LOW
 002122' 013701  021014                  MOV     @#MDMAR0,R1             ; R1 NOW CONTAINS PCBB HIGH
 002126' 010037  021010                  MOV     R0,@#MDMA0              ; POINT AT PCBB
 002132' 010137  021012                  MOV     R1,@#MDMA1
 002136' 013703  021014                  MOV     @#MDMAR0,R3             ; R3 NOW HOLDS DATA PATTERN


                                        ; ******************************************************
                                        ; ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
                                        ; ******************************************************

 002142' 012700  100000                  MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
 002146' 010067  176506                  MOV     R0,RBUFP
 002152' 005020                  10$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
 002154' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ; FILL ENTIRE BUFFER
 002160' 103774                          BLO     10$


                                        ; ********************************************
                                        ; ***** FILL XMIT BUFFER WITH TEST PATTERN *****
                                        ; ********************************************

 002162' 010067  176474                  MOV     R0,TBUFP                ; SAVE COPY OF ADDRESS
 002166' 010320                  20$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
 002170' 020027  110000                  CMP     R0,#LINADR+SIZ2K                ; STOP AT TOP
 002174' 103774                          BLO     20$


                                        ; ********************************************
                                        ; ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
```

```
                                        ; ********************************************

 002176' 012737  100200  177776                 MOV     #MODE!ENABLE,@#CMDREG   ; ENABLE LINK, SELECT MODE REG
 002204' 012737  100004  177774                 MOV     #PROM!LOOP,@#MODREG     ; PROM MODE AND LOOPBACK

 002212' 016701  176444                         MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
 002216' 005021                                 CLR     (R1)+                   ; CLEAR OUT STATUS WORD
 002220' 012721  002752                         MOV     #MAXBC-CRCSIZ,(R1)+     ; SET BYTE COUNT TO MAX ALLOWED
 002224' 005037  021034                         CLR     @#CLRFIF                ; CLEAR THE FIFO
 002230' 005067  176404                         CLR     FLG4                    ; CLEAR INTERRUPT FLAG
 002234' 016737  176420  021032                 MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUFF IS
 002242' 016737  176414  021030                 MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUFF IS

 002250' 106427  000140                         MTPS    #PRI03                  ; ALLOW XMITTER AND RECEIVER TO INTERRUPT
 002254' 032767  000100  176356  30$:           BIT     #RCVFLG,FLG4            ; WAIT FOR RECEIVER INTERRUPT
 002262' 001774                                 BEQ     30$

 002264' 032767  000040  176346  35$:           BIT     #TRNFLG,FLG4            ;WAIT FOR XMIT INTERRUPT TOO
 002272' 001774                                 BEQ     35$

 002274' 106427  000340                         MTPS    #PRI07                  ; DISABLE INTERRUPTS


                                        ; ************************************************
                                        ; ***** VERIFY THE CONTENTS OF RECEIVE BUFFER *****
                                        ; ************************************************

 002300' 012700  100000                         MOV     #LINADR,R0              ; VERIFY RECEIVE BUFFER CONTENT
 002304' 012002                                 MOV     (R0)+,R2                ; SAVE STATUS IN CASE ERROR
 002306' 005004                                 CLR     R4                      ; TRACK OFFSET IN CASE ERROR
 002310' 062700  000002                         ADD     #2,R0                   ; DON'T NEED "LENGTH" IN BUFFER

 002314' 011005                          40$:   MOV     (R0),R5                 ; READ DATA BACK
 002316' 020503                                 CMP     R5,R3                   ; R3 HOLDS ORIGINAL PATTERN
 002320' 001012                                 BNE     70$                     ; GO TO ERROR EXIT
 002322' 005200                                 INC     R0                      ; NEW ADDRESS AFTER TESTING
 002324' 005204                                 INC     R4                      ; NEXT OFFSET
 002326' 022704  002752                         CMP     #MAXBC-CRCSIZ,R4
 002332' 103770                                 BLO     40$                     ; END OF LOOP
 002334' 000241                                 CLC                             ;TELL MICROMONITOR SUCCESS

                                        ; ****************************************
                                        ; ***** FALLTHROUGH EXIT IF NO ERROR *****
                                        ; ****************************************

 002336' 112767  000005  176301  50$:           MOVB    #5,PCSR1+1              ; EXIT HERE IF NO ERROR
 002344' 000207                                 RTS     PC                      ; RETURN TO SENDER


                                        ; **********************
                                        ; ***** ERROR EXIT *****
                                        ; **********************

 002346' 016737  176274  021010  70$:           MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB ADDRESS
 002354' 016737  176270  021012                 MOV     IPCSR2+2,@#MDMA1
 002362' 013700  021014                         MOV     @#MDMAR0,R0             ; R0 NOW CONTAINS PCBB LOW
```

```
002366' 013701  021014          MOV     @#MDMAR0,R1             ; R1 NOW CONTAINS PCBB HIGH

002372' 062700  000002          ADD     #2,R0                   ; INDEX DOWN TO PCBB+2
002376' 005501                  ADC     R1

002400' 010037  021010          MOV     R0,@#MDMA0              ; POINT TO PCBB+0
002404' 010137  021012          MOV     R1,@#MDMA1

002410' 010237  021026          MOV     R2,@#MDMAW0             ; WRITE STATUS WORD TO HOST
002414' 010537  021026          MOV     R5,@#MDMAW0             ; WRITE ERRENT PATTERN TO HOST
002420' 010437  021026          MOV     R4,@#MDMAW0             ; WRITE ERROR OFFSET TO HOST

002424' 000261                  SEC                             ;TELL MICROMONITOR ERROR OCCURRED
002426' 000743                  BR      50$                     ; GO EXIT THROUGH NORMAL
```

```

                                        ; STATUS MUX VERIFICATION TEST
                                        ;
                                        ; THIS TEST INSURES THAT THE STATUS MULTIPLEXER WRITES INTO THE FIRST
                                        ; TWO LOCATIONS TO THE LINK TRANSMIT BUFFER. THE DEUNA PROCESSOR WILL
                                        ; FILL A TRANSMIT BUFFER AND A RECEIVE BUFFER AND WILL TRANSMIT A
                                        ; DATAGRAM OVER THE LOOPBACK. THE FIRST TWO WORDS OF THE RECEIVE BUFFER
                                        ; WILL BE COPIED TO HOST MEMORY FOR VERIFICATION.
                                        ;
                                        ; HOST WILL VERIFY THAT PREDETERMINED BITS OF THE STATUS WORD ARE ZERO.


                                        ; **********************************
                                        ; ***** TELL HOST WE ARE BUSY *****
                                        ; **********************************

 002430' 112767  000002  176206  MICD6:  MOVB    #INTST,PCSR1       ; TELL HOST WE ARE TESTING
 002436' 016737  176202  021020          MOV     PCSR1,@#IPCSR1


                                        ; *********************************
                                        ; ***** FILL RECEIVE BUFFER *****
                                        ; *********************************

 002444' 012703  177777                  MOV     #177777,R3         ; FILL RECEIVE BUFFER WITH ONES
 002450' 012700  100000                  MOV     #LINADR,R0         ; RECEIVE BUFFER STARTS HERE
 002454' 010067  176200                  MOV     R0,RBUFP           ; SAVE A COPY

 002460' 010320                  10$:    MOV     R3,(R0)+           ; FILL THE BUFFER
 002462' 020027  104000                  CMP     R0,#LINADR+SIZ1K
 002466' 103774                          BLO     10$

                                        ; *******************************************
                                        ; ***** FILL TRANSMIT BUFFER WITH ZERO *****
                                        ; *******************************************

 002470' 010067  176166                  MOV     R0,TBUFP           ; SAVE A COPY OF TRANSMIT ADDRESS
 002474' 005003                          CLR     R3                 ; NEED A ZERO
 002476' 010320                  20$:    MOV     R3,(R0)+           ; XMIT BUFFER STARTS 1K FROM RECEIVE
 002500' 020027  110000                  CMP     R0,#LINADR+SIZ2K
 002504' 103774                          BLO     20$


                                        ; ************************************************
                                        ; ***** FILL DATAGRAM FOR DATAGRAM LOOPBACK *****
                                        ; ************************************************

 002506' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ; ENABLE LINK MODE, SEL MODE REG
 002514' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG     ; PROMIS, ENABLE LOOPBACK

 002522' 016701  176134                  MOV     TBUFP,R1                ; POINT TO XMIT BUFFER
 002526' 005103                          COM     R3                      ; NEED SOME 1'S
 002530' 010321                          MOV     R3,(R1)+                ; BACKGROUND- SHOW STATUS OVRLAY
 002532' 012721  002752                  MOV     #MAXBC-CRCSIZ,(R1)+     ; SET BYTE COUNT TO MAX ALLOWED
 002536' 005037  021034                  CLR     @#CLRFIF                ; CLEAR THE INTERRUPT FLAG
 002542' 005067  176072                  CLR     FLG4                    ; CLEAR THE INTERRUPT FLAG
```

```
002546' 016737  176106  021032          MOV     RBUFP,@#LFRBUF          ; TELL LINK WHERE RECEIVE IS
002554' 016737  176102  021030          MOV     TBUFP,@#LTAC            ; TELL LINK WHERE XMIT IS
002562' 106427  000140                  MTPS    #PRI03                  ; WAIT FOR AN INTERRUPT
002566' 032767  000040  176044  30$:    BIT     #TRNFLG,FLG4            ; WAIT FOR XMIT INTERRUPT
002574' 001774                          BEQ     30$
002576' 032767  000100  176034  35$:    BIT     #RCVFLG,FLG4            ;WAIT FOR RECEIVER BEFORE GOING ANY FURTHER
002604' 001774                          BEQ     35$



                                ; *****************************************************
                                ; ***** GET STATUS WORDS FROM TRANSMIT BUFFER *****
                                ; *****************************************************

002606' 016700  176050                  MOV     TBUFP,R0                ; POINT AT XMIT BUFFER
002612' 012003                          MOV     (R0)+,R3                ; R5 NOW HOLDS TX0 STATUS WORD
002614' 011004                          MOV     (R0),R4                 ; R6 NOW HOLDS TX1 STATUS WORD


                                ; ***************************************************************
                                ; ***** GET HOST MEMORY ADDRESS AND WRITE STATUS WORDS *****
                                ; ***************************************************************

002616' 016737  176024  021010          MOV     IPCSR2,@#MDMA0          ; PICK UP ADDRESS OF PCBB
002624' 016737  176020  021012          MOV     IPCSR2+2,@#MDMA1
002632' 013700  021014                  MOV     @#MDMAR0,R0             ; R0=CONTENTS OF PCSR2
002636' 013702  021014                  MOV     @#MDMAR0,R2             ; R2=CONTENTS OF PCSR3
002642' 010037  021010                  MOV     R0,@#MDMA0              ; POINT AT PCBB+0
002646' 010237  021012                  MOV     R2,@#MDMA1

002652' 010337  021026                  MOV     R3,@#MDMAW0             ; WRITE TX0 TO HOST MEMORY
002656' 010437  021026                  MOV     R4,@#MDMAW0             ; WRITE TX1 TO HOST MEMORY

002662' 112767  000006  175755          MOVB    #6,PCSR1+1
002670' 000241                          CLC
002672' 000207                          RTS     PC

```

```

                                        ; LINK BYTE COUNTER TEST
                                        ;
                                        ; THIS IS "MICROCODE" FOR THE LINK BYTE COUNTER TEST.
                                        ; GETS A BYTE COUNT FROM THE HOST MEMORY. LOOPS BACK A DATAGRAM
                                        ; WITH THE BYTE COUNT TO THE RECEIVE BUFFER. VERIFIES THE RECEIVE
                                        ; BUFFER TO:
                                        ;
                                        ;       1. MAKE SURE RECEIVE BYTE COUNT IS THE SAME AS THAT DESIGNATED
                                        ;               BY THE TRANSMIT BUFFER BYTE COUNT ENTRY
                                        ;
                                        ;       2. MAKE SURE RECEIVE BUFFER WAS ACTUALLY OVERWRITTEN WITH THE
                                        ;               BYTE COUNT WRITTEN IN THE TRANSMIT BYTE COUNT BUFFER
                                        ;               ENTRY.
                                        ;
                                        ; WRITES THE VALUE OF THE RECEIVE BUFFER BYTE COUNT TO HOST MEMORY.

                                        .SBTTL  LINK BYTE COUNTER TEST

                                        ; **********************************
                                        ; ***** TELL HOST WE ARE BUSY *****
                                        ; **********************************

 002674' 112767  000002  175742  MICD7: MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
 002702' 016737  175736  021020         MOV     PCSR1,@#IPCSR1


                                        ; ********************************************************
                                        ; ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
                                        ; ********************************************************

 002710' 012700  100000                 MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
 002714' 012703  177777                 MOV     #177777,R3              ; GET ALL ONES
 002720' 010067  175734                 MOV     R0,RBUFP                ; SAVE ADDRESS OF RECV BUFFER

 002724' 010320                  10$:   MOV     R3,(R0)+                ; FILL UP THE BUFFER
 002726' 020027  104000                 CMP     R0,#LINADR+SIZ1K        ; OVERFILL- SHOW RECV OVERLAY
 002732' 103774                         BLO     10$


                                        ; ***************************************************
                                        ; ***** FILL TRANSMIT BUFFER WITH TEST PATTERN *****
                                        ; ***************************************************

 002734' 010067  175722                 MOV     R0,TBUFP                ; SAVE ADDRESS XMIT BUFFER
 002740' 005020                  20$:   CLR     (R0)+                   ; ZEROS FOR OVERLAY
 002742' 020027  110000                 CMP     R0,#LINADR+SIZ2K        ; FILL XMIT BUFFER WITH PATTERN
 002746' 103774                         BLO     20$                     ; STOP AT THE TOP


                                        ; ************************************************
                                        ; ***** RETRIEVE BYTE COUNT FROM HOST MEMORY *****
                                        ; ************************************************

 002750' 016737  175672  021010         MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB ADDRESS
 002756' 016737  175666  021012         MOV     IPCSR2+2,@#MDMA1
```

```
 002764' 013700  021014                            MOV     @#MDMAR0,R0              ; R0 NOW CONTAINS PCBB LO
 002770' 013701  021014                            MOV     @#MDMAR0,R1              ; R1 NOW CONTAINS PCBB HI
 002774' 010037  021010                            MOV     R0,@#MDMA0               ; POINT AT PCBB
 003000' 010137  021012                            MOV     R1,@#MDMA1
 003004' 013703  021014                            MOV     @#MDMAR0,R3              ; R3 NOW CONTAINS BYTE COUNT



                                        ; ********************************************
                                        ; ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
                                        ; ********************************************

 003010' 012737  100200  177776                    MOV     #MODE!ENABLE,@#CMDREG    ; ENABLE LINK, SELECT MODE REG
 003016' 012737  100014  177774                    MOV     #PROM!LOOP!DTCR,@#MODREG         ; PROM MODE AND LOOPBACK

 003024' 016704  175632                            MOV     TBUFP,R4                 ; POINT AT XMIT BUFFER
 003030' 005024                                    CLR     (R4)+                    ; CLEAR OUT STATUS WORD
 003032' 010324                                    MOV     R3,(R4)+                 ; WRITE PASSED BYTE COUNT
 003034' 005037  021034                            CLR     @#CLRFIF                 ; CLEAR THE FIFO
 003040' 005067  175574                            CLR     FLG4                     ; CLEAR INTERRUPT FLAG
 003044' 016737  175610  021032                    MOV     RBUFP,@#LFRBUF           ; TELL UNA WHERE RECIEVE BUFF IS
 003052' 016737  175604  021030                    MOV     TBUFP,@#LTAC             ; TELL UNA WHERE XMIT BUFF IS

 003060' 106427  000140                            MTPS    #PRI03                   ; ALLOW XMITTER TO INTERRUPT
 003064' 032767  000040  175546  30$:              BIT     #TRNFLG,FLG4             ; WAIT FOR INTERRUPT
 003072' 001774                                    BEQ     30$
 003074' 032767  000100  175536  35$:              BIT     #RCVFLG,FLG4             ;WAIT FOR RECEIVER INTERRUPT TOO
 003102' 001774                                    BEQ     35$


 003104' 106427  000340                            MTPS    #PRI07                   ; DISABLE INTERRUPTS



                                        ; ****************************************
                                        ; ***** COUNT ZEROS IN RECEIVE BUFFER *****
                                        ; ****************************************

 003110' 016704  175544                            MOV     RBUFP,R4                 ; VERIFY RECEIVE BUFFER CONTENTS
 003114' 062704  000002                             DD     #2,R4                    ; && COULD REPORT STATUS &&
 003120' 012402                                    MOV     (R4)+,R2                 ; SAVE "MLEN" BUFFER LENGTH
 003122' 005005                                    CLR     R5                       ; COUNT DATA WORDS TRANSFERRED

 003124' 112403                          40$:      MOVB    (R4)+,R3                 ; READ DATA BACK
 003126' 122703  000000                            CMPB    #0,R3                    ; IS IT ZERO?
 003132' 001004                                    BNE     45$                      ; EXIT IF AT THE EDGE
 003134' 005205                                    INC     R5                       ; BUMP THE TALLY
 003136' 020527  002756                            CMP     R5,#MAXBC                ; DONE YET?
 003142' 103770                                    BLO     40$                      ; IF NOT, KEEP GOING

 003144' 062700  000002                  45$:      ADD     #2,R0                    ; INDEX DOWN TO PCBB+2
 003150' 005501                                    ADC     R1

 003152' 010037  021010                            MOV     R0,@#MDMA0               ; POINT TO PCBB+2
 003156' 010137  021012                            MOV     R1,@#MDMA1

 003162' 010237  021026                            MOV     R2,@#MDMAW0              ; WRITE RECEIVE BYTE COUNT
 003166' 010537  021026                            MOV     R5,@#MDMAW0              ; WRITE BUFFER BYTE COUNT
```

```
913
914  003172' 112767  000007  175445        MOVB    #7,PCSR1+1                  ; TELL HIM WHAT TEST IT IS
915  003200' 000241                        CLC
916  003202' 000207                        RTS     PC
```

```
                                         ; LINK MEMORY ARBITRATION TEST

                                         .SBTTL  LINK MEMORY ARBITRATION TEST

                                         ; THIS IS MICROCODE FOR THE LINK ARBITRATION TEST.
                                         ; THIS MICROCODE WILL HAVE THE DMA ENGINE, THE T-11, AND BOTH STATE
                                         ; MACHINES ATTEMPTING TO ACCESS LINK MEMORY AT THE SAME TIME.
                                         ;
                                         ; EACH PROCESS OPERATES ON DATA THAT IS UNIQUE FOR IDENTIFICATION.
                                         ;
                                         ; ************************************
                                         ; ***** TELL HOST WE ARE TESTING ****
                                         ; ************************************

 003204' 112737  000002  021020  MICD8:  MOVB    #INTST,@#IPCSR1          ; TELL HOST WE ARE TESTING
 003212' 016737  175426  021020          MOV     PCSR1,@#IPCSR1

                                         ; ******************************************
                                         ; ***** RETRIEVE HOST MEMORY FROM PCBB *****
                                         ; ******************************************

 003220' 016737  175422  021010          MOV     IPCSR2,@#MDMA0           ; SET TO GET HOST PCBB ADDRESS
 003226' 016737  175416  021012          MOV     IPCSR2+2,@#MDMA1
 003234' 013704  021014                  MOV     @#MDMAR0,R4              ; R4 NOW HOLDS PCBB LOW
 003240' 013705  021014                  MOV     @#MDMAR0,R5              ; R5 NOW HOLDS PCBB HIGH

                                         ; ******************************************
                                         ; ***** FILL RECEIVE BUFFER WITH ZEROS *****
                                         ; ******************************************

 003244' 012700  100000                  MOV     #LINADR,R0               ; FILL RECEIVE BUFFER
 003250' 010067  175404                  MOV     R0,RBUFP                 ; SAVE A COPY OF POINTER
 003254' 005020                  10$:    CLR     (R0)+                    ; CLEAR IT OUT
 003256' 020027  104000                  CMP     R0,#LINADR+SIZ1K
 003262' 103774                          BLO     10$

                                         ; *****************************************
                                         ; ***** FILL TRANSMIT BUFFER WITH 33 HEX **
                                         ; ***** THATS 31463 OCTAL *****************
                                         ; *****************************************

 003264' 010067  175372                  MOV     R0,TBUFP                 ; FILL TRANSMIT BUFFER
 003270' 012720  031463          20$:    MOV     #31463,(R0)+             ; 31463 = 3333 HEX (THE REAL THING)
 003274' 020027  110000                  CMP     R0,#LINADR+SIZ2K
 003300' 103773                          BLO     20$

                                         ; ************************************
                                         ; ***** FILL DMA BUFFER WITH 0F HEX ***
```

```
                                          ; ***** THATS 7417 OCTAL **************
                                          ; **************************************

003302' 010067  175356                    MOV     R0,DBUFP                ; SAVE COPY OF POINTER
003306' 012720  007417           30$:     MOV     #7417,(R0)+
003312' 020027  114000                    CMP     R0,#LINADR+SIZ3K
003316' 103773                            BLO     30$


                                          ; **********************************
                                          ; ***** FILL BUFFER WITH FF HEX ***
                                          ; ***** THATS 177777 OCTAL ********
                                          ; **********************************

003320' 010067  175342                    MOV     R0,MBUFP                ; T-11 BUFFER POINTER
003324' 012720  177777           40$:     MOV     #177777,(R0)+
003330' 020027  120000                    CMP     R0,#LINADR+SIZ4K
003334' 103773                            BLO     40$


                                          ; **************************************
                                          ; ***** GET DMA ENGINE READY TO GO *****
                                          ; **************************************

003336' 010402                            MOV     R4,R2                   ; GET HOST ADDRESS OF PCBB
003340' 010503                            MOV     R5,R3

003342' 062702  000006                    ADD     #6,R2                   ; DUMP BUFFER AFTER ERROR CODES
003346' 005503                            ADC     R3

003350' 010237  021004                    MOV     R2,@#DMAT0              ; "TO" REGISTERS INCREMENTS
003354' 010337  021006                    MOV     R3,@#DMAT1
003360' 016737  175300  021022            MOV     DBUFP,@#DMAF            ; WHERE "FROM" REG IS LOCATED
003366' 012737  003774  021024            MOV     #3774,@#DMAWC           ; WILL XFER 1K-2 WORDS

003374' 010700                            MOV     PC,R0                   ; CALCULATE THE INTERRUPT VECTOR
003376' 062700  175074                    ADD     #DMAINT-.,R0            ; THROW IN THE OFFSET
003402' 010037  000114                    MOV     R0,@#DMAVEC
003406' 012737  000300  000116            MOV     #PRI06,@#DMAVEC+2       ;PRIORITY OF INTERRUPT SERVICE ROUTINE
003414' 005067  175236                    CLR     DMDONE                  ; FLAG

                                          ; *************************************
                                          ; ***** SET UP THE STATE MACHINES *****
                                          ; *************************************

003420' 012737  100200  177776            MOV     #MODE!ENABLE,@#CMDREG
003426' 012737  100004  177774            MOV     #PROM!LOOP,@#MODREG

003434' 016701  175222                    MOV     TBUFP,R1                ; GET ADDRESS OF XMIT BUFFER
003440' 062701  000002                    ADD     #2,R1                   ; SKIP TO BYTE COUNT
003444' 012721  002752                    MOV     #MAXBC-CRCSIZ,(R1)+     ; SET BYTE COUNT TO MAX ALLOWED
003450' 005037  021034                    CLR     @#CLRFIF                ; CLEAR INTERRUPT FLAG
003454' 005067  175160                    CLR     FLG4                    ; CLEAR INTERRUPT FLAG
003460' 016737  175174  021032            MOV     RBUFP,@#LFRBUF          ; TELL LINK WHERE RECEIVE BUF IS
003466' 016700  175174                    MOV     MBUFP,R0                ;POINT TO MICROCPU BUFFER
003472' 012702  002000                    MOV     #SIZ1K/2,R2             ;WORD COUNT FOR MICROCPU LOOP
```

```
003476' 012701  125252                          MOV     #125252,R1              ;DATA FOR MICROCPU LOOP

                                        ; ********************************
                                        ; ***** STATE THE MACHINES *****
                                        ; ********************************
                                                                        ; IS EVERYBODY READY?

003502' 016737  175154  021030                  MOV     TBUFP,@#LTAC            ; THIS STARTS THE STATE MACHINES

003510' 005237  021002                          INC     @#DMACSR                ; THIS STARTS DMA ENGINE

003514' 106427  000140                          MTPS    #PRI03                  ; ALLOW INTERRUPTS

003520' 010120                          50$:    MOV     R1,(R0)+                ;START FILLING LINK MEMORY...
003522' 077202                                  SOB     R2,50$                  ;WITH MICROCPU

                                                                        ;THE MICROCPU IS DONE

003524' 032767  000100  175106          60$:    BIT     #RCVFLG,FLG4            ;IS THE RECEIVE DONE?
003532' 001774                                  BEQ     60$                     ;NOT YET

003534' 032767  000040  175076          70$:    BIT     #TRNFLG,FLG4            ;IS THE TRANSMIT DONE?
003542' 001774                                  BEQ     70$                     ;NOT YET

003544' 005767  175106                  80$:    TST     DMDONE                  ;IS THE DMA ENGINE DONE?
003550' 001775                                  BEQ     80$                     ;NOT YET

                                                                        ;EVERYBODY IS DONE, SO NOW CHECK THE...
                                                                        ;DATA

003552' 016700  175102                          MOV     RBUFP,R0                ;POINT TO RECEIVE BUFFER
003556' 062700  000004                          ADD     #4,R0                   ;INDEX DOWN TO DATA PART
003562' 012702  001365                          MOV     #<MAXBC-CRCSIZ>/2,R2    ;AMOUNT OF DATA TO CHECK
003566' 022710  031463                  90$:    CMP     #31463,(R0)             ;IS THE DATA CORRECT?
003572' 001005                                  BNE     100$                    ;NO
003574' 062700  000002                          ADD     #2,R0                   ;YES, POINT TO NEXT WORD OF DATA
003600' 077206                                  SOB     R2,90$                  ;CONTINUE CHECKING DATA UNTIL DONE
003602' 000241                                  CLC                             ;TELL HOST THIS TEST SUCCESS
003604' 000414                                  BR      110$

003606' 010437  021010                  100$:   MOV     R4,@#MDMA0              ;POINT TO HOST PCBB
003612' 010537  021012                          MOV     R5,@#MDMA1
003616' 012737  031463  021026                  MOV     #31463,@#MDMAW0         ;GIVE HOST EXPECTED PATTERN IN PCBB
003624' 011037  021026                          MOV     (R0),@#MDMAW0           ;GIVE HOST ACTUAL PATTERN IN PCBB+2
003630' 010037  021026                          MOV     R0,@#MDMAW0             ;GIVE HOST LINK MEMORY ADDRESS
003634' 000261                                  SEC                             ;TELL MICROMONITOR ERROR OCCURRED

003636' 112767  000010  175001          110$:   MOVB    #8.,PCSR1+1             ;TELL HOST WHAT TEST THIS IS
003644' 000207                                  RTS     PC                      ;RETURN TO MICROMONITOR
```

```
                                     .SBTTL  LINK BYTE COUNTER MAXIMUM TEST
                                     ;
                                     ; ****************************************
                                     ; ***** TELL THE HOST WE ARE TESTING *****
                                     ; ****************************************
                                     ;
003646' 012767  000002  174770  MICD9:  MOV     #INTST,PCSR1            ; TELL HOST WE ARE TESTING
003654' 016737  174764  021020          MOV     PCSR1,@#IPCSR1
                                     ;
                                     ;
                                     ; ************************************************
                                     ; ***** CALCULATE BUFFER ADDRESS POINTERS *****
                                     ; ************************************************
                                     ;
003662' 012700  100000                  MOV     #LINADR,R0              ; RELATIVE TO LINK MEMORY
003666' 010067  174766                  MOV     R0,RBUFP
003672' 062700  004000                  ADD     #SIZ1K,R0
003676' 010067  174760                  MOV     R0,TBUFP
                                     ;
                                     ;
                                     ; ****************************************
                                     ; ***** SET UP LINK FOR A LOOPBACK *****
                                     ; ****************************************
                                     ;
003702' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG
003710' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG
                                     ;
003716' 016704  174740                  MOV     TBUFP,R4                ; SET UP XMIT BUFFER
003722' 005024                          CLR     (R4)+
003724' 012724  007777                  MOV     #MXMTBC,(R4)+
003730' 005037  021034                  CLR     @#CLRFIF
003734' 005067  174700                  CLR     FLG4
003740' 016737  174714  021032          MOV     RBUFP,@#LFRBUF
003746' 016737  174710  021030          MOV     TBUFP,@#LTAC
                                     ;
                                     ;
                                     ; ********************************
                                     ; ***** WAIT FOR INTERRUPTS *****
                                     ; ********************************
                                     ;
003754' 106427  000100                  MTPS    #PRI02
003760' 032767  000040  174652  10$:    BIT     #TRNFLG,FLG4
003766' 001774                          BEQ     10$
                                     ;
003770' 032767  000100  174642  20$:    BIT     #RCVFLG,FLG4
003776' 001774                          BEQ     20$
                                     ;
004000' 106427  000340                  MTPS    #PRI07
                                     ;
                                     ;
                                     ; ********************************************
                                     ; ***** CHECK RECEIVE BUFFER BYTE COUNT *****
                                     ; ********************************************
                                     ;
004004' 016704  174650                  MOV     RBUFP,R4
```

```
1135 004010' 062704  000002                 ADD     #2,R4           ; POINT AT MLEN
1136 004014' 012402                         MOV     (R4)+,R2
1137 004016' 022702  007777                 CMP     #MRECBC,R2
1138 004022' 001005                         BNE     40$
1139 004024' 000241                         CLC
1140 004026' 112767  000011  174611  30$:   MOVB    #9.,PCSR1+1
1141 004034' 000207                         RTS     PC
1142                                 ;
1143 004036' 000261                  40$:   SEC                     ; ERROR EXIT
1144 004040' 000772                         BR      30$
```

```
                                        .SBTTL  MODULE D, MICROTEST #10 - FIFO TEST

                                        ;**********************************************************************
                                        ;
                                        ;THIS IS THE MICROCODE FOR THE FIFO TEST.
                                        ;IT TRANSMITS A PACKET FROM BUFFER 0, SETS UP THE RECEIVER TO RECEIVE TO
                                        ;THE BUFFER SPECIFIED BY THE HOST IN PCBB+0. AFTER THE INTERRUPT THE BUFFER
                                        ;DONE FIFO CONTENTS ARE READ AND PASSED TO THE HOST IN PCBB+2.
                                        ;
                                        ;**********************************************************************

004042' 112767  000002  174574  MICD10: MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
004050' 016737  174570  021020          MOV     PCSR1,@#IPCSR1
                                ;
                                ;GET RECEIVE BUFFER ADDRESS FROM HOST MEMORY
                                ;
004056' 016737  174564  021010          MOV     IPCSR2,@#MDMA0          ;GET CONTENTS OF HOST'S PCSR2+3
004064' 016737  174560  021012          MOV     IPCSR2+2,@#MDMA1
004072' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 = CONTENTS OF HOST'S PCSR2
004076' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 = CONTENTS OF HOST'S PCSR3
004102' 010037  021010                  MOV     R0,@#MDMA0              ;POINT TO PCBB+0
004106' 010137  021012                  MOV     R1,@#MDMA1
004112' 013767  021014  174540          MOV     @#MDMAR0,RBUFP          ;GET RECEIVE BUFFER ADDRESS
                                ;
                                ;CLEAR ALL OF LINK MEMORY
                                ;
004120' 012702  100000                  MOV     #LINADR,R2
004124' 005022                  10$:    CLR     (R2)+
004126' 020227  177774                  CMP     R2,#LINADR+LINSIZ
004132' 103774                          BLO     10$
                                ;
                                ;FILL THE TRANSMIT BUFFER WITH 1'S
                                ;
004134' 012702  100000                  MOV     #LINADR,R2              ;USE BUFFER 0 FOR TRANSMIT
004140' 010267  174516                  MOV     R2,TBUFP                ;SAVE IT
004144' 012722  177777          20$:    MOV     #177777,(R2)+
004150' 020227  104000                  CMP     R2,#LINADR+SIZ1K
004154' 103773                          BLO     20$
                                ;
                                ;SET UP LINK FOR PROMISCUOUS MODE AND INTERNAL LOOPBACK. TRANSMIT THE MAX
                                ;SIZE PACKET. CLEAR THE FIFO AND GIVE THE LINK A RECEIVER BUFFER AND A TRANSMIT
                                ;BUFFER TO START THE OPERATION.
                                ;
004156' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ;TURN ON THE LINK AND SELECT MODE REG
004164' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG     ;SET PROMISCUOUS MODE AND LOOPBACK
004172' 016702  174464                  MOV     TBUFP,R2                ;GET TRANSMIT BUFFER
004176' 005022                          CLR     (R2)+                   ;CLEAR THE STATUS WORD
004200' 012712  002752                  MOV     #MAXBC-CRCSIZ,(R2)      ;SET TO MAX BYTE COUNT
004204' 005037  021034                  CLR     @#CLRFIF                ;CLEAR THE FIFO
004210' 005067  174424                  CLR     FLG4                    ;CLEAR THE INTERRUPT FLAG
004214' 016737  174440  021032          MOV     RBUFP,@#LFRBUF          ;LOAD THE FIFO WITH A RECEIVE BUFFER
004222' 016737  174434  021030          MOV     TBUFP,@#LTAC            ;GIVE TRANSMIT BUFFER TO START TRANSMIT
                                ;
                                ;LOWER THE PROCESSOR PRIORITY AND WAIT FOR THE INTERRUPTS
                                ;
004230' 106427  000140                  MTPS    #PRI03                  ;ALLOW BOTH TRANSMIT AND RECEIVER INTERRUPT
```

```
004234' 032767  000100  174376  30$:    BIT     #RCVFLG,FLG4            ;WAIT FOR RECEIVER FIRST
004242' 001774                          BEQ     30$

004244' 032767  000040  174366  40$:    BIT     #TRNFLG,FLG4            ;THEN THE TRANSMITTER
004252' 001774                          BEQ     40$

004254' 106427  000340                  MTPS    #PRI07                  ;DISABLE FURTHER INTERRUPTS
                                ;
                                ;READ RECEEVER BUFFER DONE FIFO AND PASS BACK TO HOST IN PCBB+2
                                ;
004260' 062700  000002                  ADD     #2,R0                   ;POINT TO PCBB+2
004264' 005501                          ADC     R1
004266' 010037  021010                  MOV     R0,@#MDMA0
004272' 010137  021012                  MOV     R1,@#MDMA1
004276' 013737  021044  021026          MOV     @#LRBUF,@#MDMAW0        ;PASS BUFFER DONE DATA TO HOST

004304' 112767  000012  174333          MOVB    #10.,PCSR1+1            ;INDICATE WHAT TEST WE JUST FINISHED
004312' 000241                          CLC                             ;INDICATE SUCCESS
004314' 000207                          RTS     PC

```

```
                                        .SBTTL  MODULE D, MICROTEST #11 - LINK ADDRESS TEST

                                        ;*************************************************************************
                                        ;
                                        ;THIS THE MICROCODE FOR THE RECEIVER AND TRANSMITTER LINK MEMORY ADDRESS TESTS.
                                        ;IT FILLS ALL OF LINK MEMORY WITH 0'S THEN FILLS THE TRANSMIT BUFFER OBTAINED
                                        ;FROM THE HOST'S PCBB+2 WITH 1'S. IT SETS UP THE RECEIVER TO RECEIVE A PACKET
                                        ;INTO THE BUFFER OBTAINED FROM THE HOST'S PCBB+0. THE PACKET OF 1'S IS
                                        ;TRANSMITTED WITH LOOPBACK ENABLED IN PROMISCUIOUS MODE. AFTER THE INTERRUPT
                                        ;ALL LINK MEMORY OUTSIDE OF THE TRANSMIT AND RECEIVE BUFFERS IS CHECKED
                                        ;TO SEE IF AND DATA WAS WRITTEN TO AN INCORRECT ADDRESS. IF SO THE FAULTY
                                        ;ADDRESS IS PASSED BACK TO THE HOST IN PCBB+4 AND THE GOOD DATA IN PCBB+6 AND
                                        ;THE BAD DATA IN PCBB+10.
                                        ;
                                        ;*************************************************************************

004316' 112767  000002  174320  MICD11: MOVB    #INTST,PCSR1            ;TELL HOST WE ARE TESTING
004324' 016737  174314  021020          MOV     PCSR1,@#IPCSR1

                                        ;
                                        ;GET THE RECEIVER BUFFER FROM THE HOST'S PCBB+0 AND THE TRANSMIT BUFFER FROM
                                        ;THE HOST'S PCBB+2
                                        ;
004332' 016737  174310  021010          MOV     IPCSR2,@#MDMA0          ;GET CONTENTS OF HOST'S PCSR2+3
004340' 016737  174304  021012          MOV     IPCSR2+2,@#MDMA1
004346' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 = CONTENTS OF HOST'S PCSR2
004352' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 = CONTENTS OF HOST'S PCSR3
004356' 010037  021010                  MOV     R0,@#MDMA0              ;POINT TO PCBB+0
004362' 010137  021012                  MOV     R1,@#MDMA1
004366' 013767  021014  174264          MOV     @#MDMAR0,RBUFP          ;GET RECEIVER BUFFER FROM PCBB+0
004374' 013767  021014  174260          MOV     @#MDMAR0,TBUFP          ;GET TRANSMIT BUFFER FROM PCBB+2
                                        ;
                                        ;FILL ALL OF LINK MEMORY WITH 0'S
                                        ;
004402' 012702  100000                  MOV     #LINADR,R2
004406' 005022                  10$:    CLR     (R2)+
004410' 020227  177774                  CMP     R2,#LINADR+LINSIZ
004414' 103774                          BLO     10$

004416' 016702  174240                  MOV     TBUFP,R2                ;POINT TO BASE OF TRANSMIT BUFFER
004422' 005022                          CLR     (R2)+                   ;CLEAR STATUS WORD
004424' 012722  002756                  MOV     #MAXBC,(R2)+            ;PUT IN BYTE COUNT
                                        ;
                                        ;FILL TRANSMIT BUFFER WITH 1'S
                                        ;
004430' 012701  002756                  MOV     #MAXBC,R1
004434' 006201                          ASR     R1
004436' 012722  177777          20$:    MOV     #177777,(R2)+
004442' 005301                          DEC     R1
004444' 001374                          BNE     20$

                                        ;
                                        ;SET UP LINK FOR PROMISCIOUS MODE AND INTERNAL LOOPBACK. TRANSMIT THE PACKET
                                        ;
004446' 012737  000200  177776          MOV     #MODE,@#CMDREG          ;SELECT THE MODE REGISTER
004454' 012737  100014  177774          MOV     #PROM!LOOP!DTCR,@#MODREG ;ENABLE PROMISCUIOUS MODE AND LOOPBACK
```

```
                                                                ;AND DISABLE TRANSMIT CRC
004462' 012737  100000  177776          MOV     #ENABLE,@#CMDREG        ;TURN ON THE LINK
004470' 005037  021034                  CLR     @#CLRFIF                ;CLEAR THE FIFO
004474' 005067  174140                  CLR     FLG4                    ;CLEAR THE INTERRUPT WORD
004500' 016737  174154  021032          MOV     RBUFP,@#LFRBUF          ;GIVE THE LINK A RECEIVE BUFFER
004506' 016737  174150  021030          MOV     TBUFP,@#LTAC            ;GIVE THE LINK A TRANSMIT BUFFER
                                                                ;TO START THE OPERATION
                                ;
                                ;LOWER THE PROCESSOR PRIORITY AND WAIT FOR THE INTERRUPT
                                ;
004514' 106427  000140                  MTPS    #PRI03                  ;ALLOW BOTH TRANSMIT AND RECEIVE INTERRUPT
004520' 032767  000100  174112  30$:    BIT     #RCVFLG,FLG4            ;WAIT FOR THE RECEIVE INTERRUPT
004526' 001774                          BEQ     30$
004530' 032767  000040  174102  40$:    BIT     #TRNFLG,FLG4
004536' 001774                          BEQ     40$

004540' 106427  000340                  MTPS    #PRI07                  ;DISABLE FURTHER INTERRUPTS
                                ;
                                ;CHECK RECEIVE BUFFER TO MAKE SURE IT RECEIVED NON-ZERO DATA
                                ;
004544' 012703  177777                  MOV     #177777,R3              ;PASS THIS NON-ZERO DATA IF FAILURE
004550  016702  174104                  MOV     RBUFP,R2                ;GET POINTER TO RECEIVER BUFFER
004554' 062702  000004                  ADD     #4,R2                   ;SKIP STATUS AND BYTE COUNT
004560' 012701  002756                  MOV     #MAXBC,R1               ;NUMBER OF BYTES WE SENT
004564' 006201                          ASR     R1                      ;MAKE IT WORDS
004566' 005712                  45$:    TST     (R2)                    ;IS THERE NON-ZERO DATA HERE?
004570' 001432                          BEQ     90$                     ;NO, ERROR!
004572' 062702  000002                  ADD     #2,R2                   ;POINT TO NEXT WORD IN RECEIVER BUFFER
004576' 005301                          DEC     R1
004600' 001372                          BNE     45$
                                ;
                                ;CHECK ALL LINK MEMORY EXCEPT THE TRANSMIT AND RECEIVE BUFFERS FOR ANY NON-ZERO
                                ;DATA
                                ;
004602' 005003                          CLR     R3                      ;PASS THIS ZERO DATA IF FAILURE
004604' 012702  100000                  MOV     #LINADR,R2              ;START AT BASE OF LINK MEMORY
004610' 020267  174046          50$:    CMP     R2,TBUFP                ;ARE WE AT THE TRANSMIT BUFFER?
004614' 001002                          BNE     60$                     ;NO
004616' 062702  002762                  ADD     #MAXBC+4,R2             ;YES, SKIP OVER THE HEADER AND THE DATA
004622' 020267  174032          60$:    CMP     R2,RBUFP                ;ARE WE AT THE RECEIVE BUFFER?
004626' 001002                          BNE     70$                     ;NO
004630' 062702  002762                  ADD     #MAXBC+4,R2             ;YES, SKIP OVER THE HEADER AND THE DATA
004634' 005712                  70$:    TST     (R2)                    ;IS ANY NON-ZERO DATA IN HERE?
004636' 001007                          BNE     90$                     ;YES, ERROR!
004640' 062702  000002                  ADD     #2,R2                   ;POINT TO NEXT WORD IN LINK MEMORY
004644' 020227  177774                  CMP     R2,#LINADR+LINSIZ       ;HAVE WE CHECKED ALL OF LINK MEMORY?
004650' 103757                          BLO     50$                     ;NOT YET
                                ;
                                ;TEST WAS SUCESSFULL
                                ;
004652' 000241                          CLC                             ;INDICATE SUCCESS
004654' 000407                          BR      100$                    ;LEAVE
                                ;
                                ;PASS THE ADDRESS BACK TO THE HOST IN PCBB+4 AND THE DATA IN PCBB+6
                                ;
004656' 010237  021026          90$:    MOV     R2,@#MDMAWO             ;PASS ADDRESS TO PCBB+4
```

```
004662' 010337  021026                   MOV     R3,@#MDMAWO             ;PASS GOOD DATA TO PCBB+6
004666' 011237  021026                   MOV     (R2),@#MDMAWO           ;PASS DATA TO PCBB+10
004672' 000261                           SEC                             ;INDICATE FAILURE

004674' 112767  000013  173743  100$:    MOVB    #11.,PCSR1+1            ;TELL HOST WHAT TEST JUST FINISHED
004702' 000207                           RTS     PC


004704' 004706                  MICDSZ::MICDSZ-MICROD+2
        000001                  .END
```

```
ACLO   = 002000              87#
ADRERR= 000001              132#
ADRREG= 177774               34#
ARAM   = 000100              80#
BIT0   = 000001              54#    103
BIT1   = 000002              53#    104
BIT10  = 002000              44#     56      73      87
BIT11  = 004000              43#     72      86
BIT12  = 010000              42#     56      71     113
BIT13  = 020000              41#     70     112
BIT14  = 040000              40#     69      85     111     240
BIT15  = 100000              39#     68      78      84     110     245     248
BIT2   = 000004              52#     91     105
BIT3   = 000010              51#     90     106
BIT4   = 000020              50#     89     107
BIT5   = 000040              49#     88     108
BIT6   = 000100              48#     80     109
BIT7   = 000200              47#     79
BIT8   = 000400              46#     56      74
BIT9   = 001000              45#
CLRFIF= 021034               22#    309*    342*    405*    519*    644*    761*    874*   1024*   1110*   1193*   1279*
CMDREG= 177776               35#    138*    303*    336*    399*    513*    638*    754*    868*   1018*   1104*   1188*   1275*
                           1278*
COLL   = 000020              89#
CRCSIZ= 000004              130#    308     341     404     434     518     550     643     673     760    1023    1061    1192
CSRFLG= 000001              103#    196     227
CSRVEC= 000064               96#    163*    164*
CSRWRT   000436R     002    162     227#
DATERR= 000000              131#
DBUFP    000664R     002    286#    976*   1005
DMACSR= 021002                9#    239     248*   1038*
DMAF   = 021022              17#   1005*
DMAINT   000472R     002    157     239#   1009
DMAT0  = 021004              10#   1003*
DMAT1  = 021006              11#   1004*
DMAVEC= 000114               97#    158*    159*   1010*   1011*
DMAWC  = 021024              18#   1006*
DMDONE   000656R     002    239*    240     245     283#   1012*   1053
DNI    = 004000              72#    142     220
DRTY   = 000040              88#
DTCR   = 000010              90#    869    1276
ENABLE= 100000               78#    303     336     399     513     638     754     868    1018    1104    1188    1278
ENAL   = 040000              85#
ENCR   = 004000              86#
ERRFLG= 000002              104#    230
ERRINT   000446R     002    144     230#
FATI   = 000400              74#
FLG4     000640R     002    189*    192     196     221*    227*    230*    253*    260*    263*    278#    310*    315     343*
                            348     352     406*    411     414     520*    525     528     645*    650     653     762*    766
                            768     875*    880     882    1025*   1047    1050    1111*   1121    1124    1194*   1201    1204
                           1280*   1288    1290
INERR  = 000003             102#    218
INMON  = 000001             100#    140     204     216
INTST  = 000002             101#    289     322     370     482     597     723     824     935    1086    1156    1237
IOADR  = 020000             124#    125
IOSIZ  = 020000             120#    125
```

```
IPCSR0= 021000          8#    142*    201     220*    231*    242*    247*    254*
IPCSR1= 021020         16#    141*    204*    219*    290*    323*    371*    483*    598*    724*    825*    935*    936*
                     1087*   1157*   1238*
IPCSR2  000646R   002  187*    188*    281#    450     451     565     566     604     605     689     690     786     787
                      855     856     943     944    1161    1162    1244    1245
LASFTP= 012400         56#
LFRBUF= 021032         21#    311*    344*    407*    521*    646*    763*    876*   1026*   1112*   1195*   1281*
LINADR= 100000        126#    292     295     300     325     328     333     377     380     391     424     490     493
                      505     538     617     620     630     663     732     736     746     832     837     847     953
                      956     967     978     989    1094    1171    1173    1178    1181    1255    1257    1312    1322
LINSIZ= 077774        122#   1173    1257    1322
LOOP  = 000004         91#    304     337     400     514     639     755     869    1019    1105    1189    1276
LRBUF = 021044         26#    258    1215
LTAC  = 021030         20#    312*    345*    408*    522*    647*    764*    877*   1036*   1113*   1196*   1282*
MAXBC = 002756        127#    308     341     404     434     518     550     643     673     760     902    1023    1061
                     1192    1262    1266    1300    1315    1318
MBUFP   000666R   002  287#    987*   1027
MDMAR0= 021014         14#    452     453     567     568     606     607     610     691     692     788     789     857
                      858     861     945     946    1163    1164    1167    1246    1247    1250    1251
MDMAR1= 021016         15#
MDMAW0= 021026         19#    457*    458*    459*    460*    572*    573*    574*    575*    700*    701*    702*    793*
                      794*    911*    912*   1071*   1072*   1073*   1215*   1332*   1333*   1334*
MDMAW1= 021036         23#
MDMA0 = 021010         12#    450*    454*    565*    569*    604*    608*    689*    697*    786*    790*    855*    859*
                      908*    943*   1069*   1161*   1165*   1213*   1244*   1248*
MDMA1 = 021012         13#    451*    455*    566*    570*    605*    609*    690*    698*    787*    791*    856*    860*
                      909*    944*   1070*   1162*   1166*   1214*   1245*   1249*
MDMSR = 021042         25#
MICDSZ  004704RG  002 1341#
MICD1   000670R   002  266     289#
MICD10  004042R   002  275    1156#
MICD11  004316R   002  276    1237#
MICD2   001052R   002  267     322#
MICD3   001250R   002  268     370#
MICD4   001554R   002  269     482#
MICD5   002066R   002  270     597#
MICD6   002430R   002  271     723#
MICD7   002674R   002  272     824#
MICD8   003204R   002  273     935#
MICD9   003646R   002  274    1086#
MICROD  000000RG  002  137#   1341
MODE  = 000200         79#    303     336     399     513     638     754     868    1018    1104    1188    1275
MODREG= 177774         33#    304*    337*    400*    514*    639*    755*    869*   1019*   1105*   1189*   1276*
MRECBC= 007777        129#   1137
MXMTBC= 007777        128#   1109
NPRERR= 100000        110#    247
NPRFLG= 000020        107#
NXMERR= 040000        111#    242
NXMFLG= 000010        106#
PARERR= 010000        113#    254
PARFLG= 000004        105#    253
PARINT  000550R   002  152     253#
PARVEC= 000140         98#    153*    154*
PCBADR  000652R   002  282#
PCEI  = 040000         69#
PCSRSW= 021040         24#    181
```

```
PCSR1    000644R   002    140*    141     216*    218*    219     280#    289*    290     318*    322*    323     356*    370*
                          371     442*    482*    483     558*    597*    598     681*    723*    724     796*    824*    825
                          914*    936     1076*   1086*   1087    1140*   1156*   1157    1217*   1237*   1238    1337*
PHYAD0= 021060             27#
PHYAD1= 021062             28#
PHYAD2= 021064             29#
PHYAD3= 021066             30#
PHYAD4= 021070             31#
PHYAD5= 021072             32#
PRIO0 = 000000             64#    190*
PRIO1 = 000040             63#
PRIO2 = 000100             62#    1120*
PRIO3 = 000140             61#    314*    351*    410*    524*    649*    765*    879*    1040*   1200*   1287*
PRIO4 = 000200             60#    164     174     347*
PRIO5 = 000240             59#    169     179
PRIO6 = 000300             58#    1011
PRIO7 = 000340             57#    137*    147     154     159     195*    317*    355*    417*    531*    656*    886*    1127*
                          1207*   1293*
PROM  = 100000             84#    304     337     400     514     639     755     869     1019    1105    1189    1276
RBUFP    000660R   002    284#    293*    311     326*    344     378*    407     491*    521     618*    646     733*    763
                          834*    876     893     954*    1026    1059    1095*   1112    1134    1167*   1195    1250*   1281
                          1298    1316
RCEI  = 002000             73#
RCVFLG= 000100            109#    260     348     411     525     650     768     882     1047    1124    1201    1288
RCVINT   000566R   002    177     258#
RCVVEC= 000120             94#    178*    179*
ROMADR= 040000            125#    126
ROMSIZ= 040000            121#    126
RXI   = 020000             70#
SANTIM   000642R   002    236*    279#
SANVEC= 000134             95#    168*    169*
SERI  = 100000             68#
SIZ1K = 004000            114#    115     116     295     328     380     493     620     736     837     956     1028    1096
                          1181
SIZ2K = 010000            115#    117     300     333     391     505     630     746     847     967
SIZ3K = 014000            116#    978
SIZ4K = 020000            117#    118     119     120     989
SIZ8K = 040000            118#    121     122
STACK = 001000             99#    139
TBLD     000612R   002    210     266#
TBUFP    000662R   002    285#    298*    306     312     331*    339     345     389*    402     408     502*    516     522
                          628*    641     647     743*    757     764     777     845*    871     877     965*    1021    1036
                          1097*   1107    1113    1179*   1190    1196    1251*   1260    1282    1313
TIMINT   000464R   002    167     236#
TRNFLG= 000040            108#    263     315     352     414     528     653     766     880     1050    1121    1204    1290
TRNINT   000602R   002    172     263#
TRNVEC= 000070             93#    173*    174*
TXI   = 010000             71#
UNIERR= 020000            112#    231
WCSADR= 000000            123#    124
WCSSIZ= 020000            119#    124
.     = 004706R    002    144     152     157     162     167     172     177     198     210     233     249     256     266
                          267     268     269     270     271     272     273     274     275     276     1009
```

```
BCOMPL      1#
BERROR      1#
BGNAU       1#
BGNAUT      1#
BGNCLN      1#
BGNDU       1#
BGNHRD      1#
BGNHW       1#
BGNINI      1#
BGNMOD      1#
BGNMSG      1#
BGNPRO      1#
BGNPTA      1#
BGNRPT      1#
BGNSEG      1#
BGNSET      1#
BGNSFT      1#
BGNSRV      1#
BGNSUB      1#
BGNSW       1#
BGNTST      1#
BNCOMP      1#
BNERRO      1#
BREAK       1#
BRESET      1#
CKLOOP      1#
CLOCK       1#
CLOSE       1#
CLRVEC      1#
COMMEN      1#
DELAY       1#
DESCRI      1#
DEVTYP      1#
DISPAT      1#
DISPLA      1#
DOCLN       1#
DODU        1#
DORPT       1#
ENDAU       1#
ENDAUT      1#
ENDCLN      1#
ENDCOM      1#
ENDDU       1#
ENDHRD      1#
ENDHW       1#
ENDINI      1#
ENDMOD      1#
ENDMSG      1#
ENDPRO      1#
ENDPTA      1#
ENDRPT      1#
ENDSEG      1#
ENDSET      1#
ENDSFT      1#
ENDSRV      1#
ENDSUB      1#
```

```
ENDSW      1#
ENDTST     1#
EQUALS     1#
ERRDF      1#
ERRHRD     1#
ERROR      1#
ERRSF      1#
ERRSOF     1#
ERRTBL     1#
ESCAPE     1#
EXIT       1#
FEQUAL     1#
GETBYT     1#
GETPRI     1#
GETWOR     1#
GMANIA     1#
GMANID     1#
GMANIL     1#
GPHARD     1#
GPRMA      1#
GPRMD      1#
GPRML      1#
HEADER     1#
INLOOP     1#
IOSETU     1#
IOSTAR     1#
KT11       1#
LASTAD     1#
MANUAL     1#
MEMORY     1#
MSBYTE     1#
MSCHEC     1#
MSCNTO     1#
MSCOUN     1#
MSDATA     1#
MSDECR     1#
MSDEFA     1#
MSENDE     1#
MSERRI     1#
MSESCA     1#
MSESCS     1#
MSEXCP     1#
MSEXIT     1#
MSEXSE     1#
MSEXTJ     1#
MSGEN      1#
MSGENB     1#
MSGETS     1#
MSGETT     1#
MSGNGB     1#
MSGNIN     1#
MSGNLS     1#
MSGNSU     1#
MSGNTA     1#
MSGNTE     1#
MSHAPT     1#
```

```
MSHNAP       1#
MSINCR       1#
MSIOSE       1#
MSLDRO       1#
MSMASK       1#
MSMCHI       1#
MSMCLO       1#
MSMSK1       1#
MSPOP        1#
MSPRIN       1#
MSPUSH       1#
MSPUT        1#
MSPUT1       1#
MSRADI       1#
MSRBRO       1#
MSRNRO       1#
MSSETS       1#
MSSTAR       1#
MSSVC        1#
MSTLAB       1#
MSTSTL       1#
MSWORD       1#
MSXFER       1#
OPEN         1#
POINTE       1#
PRINTB       1#
PRINTF       1#
PRINTS       1#
PRINTX       1#
READBU       1#
READEF       1#
RFLAGS       1#
SETPRI       1#
SETVEC       1#
SLASH        1#
STARS        1#
SVC          1#
XFER         1#
XFERF        1#
XFERT        1#

. ABS.  000000      000
        000000      001
MICRD   004706      002

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROD.OBJ,MICROD.LST/CR/SOL/NL:TOC=SVC34R.P11,MICROD.MAC
 RUN-TIME: 3 4 .4 SECONDS
 RUN-TIME RATIO: 39/7=5.0
 CORE USED:  31K  (61 PAGES)
```

```
                        .TITLE  MICROE - MICROCODE MODULE E
                        ; && DEDICATED THE CRC CIRCUITRY TO THE RECEIVE SIDE OF THE LINK

        000000'         .CSECT MICRE

                        .SBTTL   REGISTER DEFINITIONS USED BY THE T11

        021000          IPCSR0  =       21000   ;INTERNAL PCSR0 ADDRESS
        021002          DMACSR  =       21002   ;DMA ENGINE CONTROL STATUS REGISTER
        021004          DMAT0   =       21004   ;DMA ENGINE TO ADDRESS REGISTER #0
        021006          DMAT1   =       21006   ;DMA ENGINE TO ADDRESS REGISTER #1
        021010          MDMA0   =       21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
        021012          MDMA1   =       21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
        021014          MDMAR0  =       21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
        021016          MDMAR1  =       21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
        021020          IPCSR1  =       21020   ;INTERNAL PCSR1 ADDRESS
        021022          DMAF    =       21022   ;DMA ENGINE FROM ADDRESS REGISTER
        021024          DMAWC   =       21024   ;DMA ENGINE WORD COUNT REGISTER
        021026          MDMAW0  =       21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
        021030          LTAC    =       21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
        021032          LFRBUF  =       21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
        021034          CLRFIF  =       21034   ;CLEAR FIFO
        021036          MDMAW1  =       21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
        021040          PCSRSW  =       21040   ;SWITCH PACK REGISTER
        021042          MDMSR   =       21042   ;MICROCPU DMA STATUS REGISTER
        021044          LRBUF   =       21044   ;LINK RECIEVE BUFFER COMPLETED
        021060          PHYAD0  =       21060   ;PHYSICAL ADDRESS ROM BYTE 0
        021062          PHYAD1  =       21062   ;PHYSICAL ADDRESS ROM BYTE 1
        021064          PHYAD2  =       21064   ;PHYSCIAL ADDRESS ROM BYTE 2
        021066          PHYAD3  =       21066   ;PHYSICAL ADDRESS ROM BYTE 3
        021070          PHYAD4  =       21070   ;PHYSICAL ADDRESS ROM BYTE 4
        021072          PHYAD5  =       21072   ;PHYSICAL ADDRESS ROM BYTE 5
        177774          MODREG  =       177774  ;LINK MODE REGISTER
        177774          ADRREG  =       177774  ;LINK STATION ADDRESS RAM REGISTER
        177776          CMDREG  =       177776  ;LINK COMMAND REGISTER

                        .SBTTL   OTHER DEFINITIONS USED BY THE MICROCODE

        100000          BIT15   =       100000
        040000          BIT14   =       40000
        020000          BIT13   =       20000
        010000          BIT12   =       10000
        004000          BIT11   =       4000
        002000          BIT10   =       2000
        001000          BIT9    =       1000
        000400          BIT8    =       400
        000200          BIT7    =       200
        000100          BIT6    =       100
        000040          BIT5    =       40
        000020          BIT4    =       20
        000010          BIT3    =       10
        000004          BIT2    =       4
        000002          BIT1    =       2
        000001          BIT0    =       1
                        ;
        012400          LASFTP  =       BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
```

```
        000340                  PRI07   =       340
        000300                  PRI06   =       300
        000240                  PRI05   =       240
        000200                  PRI04   =       200
        000140                  PRI03   =       140
        000100                  PRI02   =       100
        000040                  PRI01   =       40
        000000                  PRI00   =       0
                                ;
                                ;PCSR0 - PORT CONTROL STATUS REGISTER 0
                                ;
        100000                  SERI    =       BIT15
        040000                  PCEI    =       BIT14
        020000                  RXI     =       BIT13
        010000                  TXI     =       BIT12
        004000                  DNI     =       BIT11
        002000                  RCEI    =       BIT10
        000400                  FATI    =       BIT8
                                ;
                                ;LINK COMMAND REGISTER
                                ;
        100000                  ENABLE  =       BIT15   ;ENABLE LINK MODULE
        000200                  MODE    =       BIT7    ;ENABLE MODE REGISTER
        000100                  ARAM    =       BIT6    ;ENABLE STATION ADDRESS RAM
                                ;
                                ;LINK MODE REGISTER
                                ;
        100000                  PROM    =       BIT15   ;PROMISCUIOUS MODE
        040000                  ENAL    =       BIT14   ;ENABLE MULTICAST
        004000                  ENCR    =       BIT11   ;ENABLE COLLISION TEST
        002000                  ACLO    =       BIT10   ;ENABLE ACLO
        000040                  DRTY    =       BIT5    ;DISABLE RETRY LOGIC
        000020                  COLL    =       BIT4    ;SIMULATE A COLLISION
        000010                  DTCR    =       BIT3    ;DISABLE TRANSMIT CRC LOGIC
        000004                  LOOP    =       BIT2    ;ENABLE LOOPBACK

        000070                  TRNVEC= 70              ;VECTOR ADDRESS FOR THE TRANSMITTER
        000120                  RCVVEC= 120             ;VECTOR ADDRESS FOR THE RECEIVER
        000134                  SANVEC= 134             ;VECTOR ADDRESS FOR THE SANITY TIMER
        000064                  CSRVEC= 64              ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
        000114                  DMAVEC= 114             ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
        000140                  PARVEC= 140             ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
        001000                  STACK=  1000            ;STACK LOCATION
        000001                  INMON=  1               ;IN MICROMONITOR STATE
        000002                  INTST=  2               ;IN A TEST STATE
        000003                  INERR=  3               ;IN ERROR STATE
        000001                  CSRFLG= BIT0            ;CSR WRITE INTERRUPT OCCURED
        000002                  ERRFLG= BIT1            ;UNEXPECTED ERROR OCCURED
        000004                  PARFLG= BIT2            ;PARITY ERROR OCCURED
        000010                  NXMFLG= BIT3            ;NON-EXISTANT MEMORY ERROR OCCURRED
        000020                  NPRFLG= BIT4            ;NPR TIMEOUT OCCURRED
        000040                  TRNFLG= BIT5            ;TRANSMITTER INTERRUPT OCCURRED
        000100                  RCVFLG= BIT6            ;RECEIVER INTERRUPT OCCURRED
        100000                  NPRERR= BIT15           ;PCSR0 FLAG INDICATING NPR ERROR OCCURRED
        040000                  NXMERR= BIT14           ;PCSR0 FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
        020000                  UNIERR= BIT13           ;PCSR0 FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
```

```
        010000          PARERR= BIT12           ;PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
        020000          MTCH=   BIT13           ;MATCH BIT
        004000          SIZ1K=  4000            ;1K WORDS
        010000          SIZ2K=  SIZ1K*2         ;2K WORDS
        020000          SIZ4K=  SIZ2K*2         ;4K WORDS
        040000          SIZ8K=  SIZ4K*2         ;8K WORDS
        020000          WCSSIZ= SIZ4K           ;SIZE OF WRITEABLE CONTROL STORE
        020000          IOSIZ=  SIZ4K           ;SIZE OF I/O PAGE
        040000          ROMSIZ= SIZ8K           ;SIZE OF ROM
        077774          LINSIZ= SIZ8K*2-4       ;SIZE OF LINK MEMORY
        000000          WCSADR= 0               ;BASE ADDRESS OF WCS
        020000          IOADR=  WCSADR+WCSSIZ   ;BASE ADDRESS OF I/O PAGE
        040000          ROMADR= IOADR+IOSIZ     ;BASE ADDRESS OF ROM
        100000          LINADR= ROMADR+ROMSIZ   ;BASE ADDRESS OF LINK MEMORY
        000100          MINBC=  64.             ; 64 BYTES
        002752          MAXBC=  1518.-4.        ;MAXIMUM # OF BYTES IN A TRANSMIT BUFFER
        000000          DATERR= 0               ;FLAG INDICATING DATA ERROR OCCURRED
        000001          ADRERR= 1               ;FLAG INDICATING ADDRESS ERROR OCCURRED
```

```
                                          .SBTTL  E_MODULE MICROCODE

000000' 106427  000340                    MICROE::MTPS    #PRI07                  ;DISABLE INTERRUPTS
000004' 012737  000000  177776                    MOV     #0,@#CMDREG             ;TURN OFF THE LINK
000012' 012706  001000                            MOV     #STACK,SP               ;SETUP STACK
000016' 112767  000001  000602                    MOVB    #INMON,PCSR1            ;TELL HOST WE ARE IN MICROMONITOR
000024' 016737  000576  021020                    MOV     PCSR1,@#IPCSR1
000032' 012737  004000  021000                    MOV     #DNI,@#IPCSR0           ;TELL HOST THE LOAD AND START FINISHED
000040' 010700                                    MOV     PC,R0                   ;GET ADDRESS OF UNEXPECTED ERROR...
000042' 062700  000404                            ADD     #ERRINT-.,R0            ;HANDLER
000046' 005001                                    CLR     R1                      ;FILL ALL UNUSED VECTORS WITH TRAP...
000050' 010021                            10$:    MOV     R0,(R1)+                ;HANDLER
000052' 012721  000340                            MOV     #PRI07,(R1)+
000056' 020127  001000                            CMP     R1,#1000
000062' 002772                                    BLT     10$

000064' 010700                                    MOV     PC,R0                   ;SETUP PARITY TRAP VECTOR
000066' 062700  000462                            ADD     #PARINT-.,R0
000072' 010037  000140                            MOV     R0,@#PARVEC
000076' 012737  000340  000142                    MOV     #PRI07,@#PARVEC+2

000104' 010700                                    MOV     PC,R0                   ;SETUP DMA INTERRUPT VECTOR
000106' 062700  000364                            ADD     #DMAINT-.,R0
000112' 010037  000114                            MOV     R0,@#DMAVEC
000116' 012737  000340  000116                    MOV     #PRI07,@#DMAVEC+2

000124' 010700                                    MOV     PC,R0                   ;SETUP CSR WRITE VECTOR
000126' 062700  000310                            ADD     #CSRWRT-.,R0
000132' 010037  000064                            MOV     R0,@#CSRVEC
000136' 012737  000200  000066                    MOV     #PRI04,@#CSRVEC+2

000144' 010700                                    MOV     PC,R0                   ;SETUP SANTITY TIMER VECTOR
000146' 062700  000316                            ADD     #TIMINT-.,R0
000152' 010037  000134                            MOV     R0,@#SANVEC
000156' 012737  000240  000136                    MOV     #PRI05,@#SANVEC+2

000164' 010700                                    MOV     PC,R0                   ;SETUP TRANSMITTER VECTOR
000166' 062700  000414                            ADD     #TRNINT-.,R0
000172' 010037  000070                            MOV     R0,@#TRNVEC
000176' 012737  000200  000072                    MOV     #PRI04,@#TRNVEC+2

000204' 010700                                    MOV     PC,R0                   ;SETUP RECEIVER VECTOR
000206' 062700  000360                            ADD     #RCVINT-.,R0
000212' 010037  000120                            MOV     R0,@#RCVVEC
000216' 012737  000240  000122                    MOV     #PRI05,@#RCVVEC+2

000224' 013700  021040                            MOV     @#PCSRSW,R0             ;GET SWITCH PACK BITS
000230' 052700  176000                            BIS     #176000,R0              ;MAP THEM INTO HOST I/O PAGE
000234' 006300                                    ASL     R0                      ;SHIFT OVER TO POSITION CORRECTLY
000236' 006300                                    ASL     R0
000240' 006300                                    ASL     R0
000242' 062700  000004                            ADD     #4,R0                   ;PCSR2 IS PCSR0+4
000246' 010067  000356                            MOV     R0,IPCSR2               ;SAVE PCSR2 ADDRESS
000252' 012767  000003  000352                    MOV     #3,IPCSR2+2             ;HIGH ORDER BITS 17:16
000260' 005067  000336                            CLR     FLG5                    ;INITIALIZE FLAG WORD
```

```
000264' 106427  000000          15$:    MTPS    #PRI00          ;ALLOW INTERRUPTS

000270' 005767  000326          20$:    TST     FLG5            ;WAIT FOR A COMMAND FROM HOST
000274' 001775                          BEQ     20$

000276' 106427  000340                  MTPS    #PRI07          ;RAISE CPU PRIORITY TO SERVICE COMMAND
000302' 032767  000001  000312          BIT     #CSRFLG,FLG5    ;DID HOST GIVE US A COMMAND?
000310' 001001                          BNE     30$             ;YES
000312' 000777                          BR      .               ;NO, ERROR SO JUST SIT HERE...
                                                                ;FOR LACK OF ANYTHING BETTER TO DO

000314' 113700  021000          30$:    MOVB    @#IPCSR0,R0     ;GET WHAT HOST WROTE TO PCSR0
000320' 042700  177760                  BIC     #177760,R0      ;STRIP ALL BUT COMMAND BITS
000324' 001004                          BNE     35$             ;WAS IT THE CLEAR FUNCTION?
000326' 012737  000001  021020          MOV     #INMON,@#IPCSR1 ;YES, CLEAR OUT THE TEST # BITS
000334' 000432                          BR      50$
000336' 022700  000017          35$:    CMP     #17,R0          ;RESTART OPERATIONAL MICROCODE?
000342' 001432                          BEQ     60$             ;YES
000344' 162700  000001                  SUB     #1,R0
000350' 010701                          MOV     PC,R1           ;GET ADDRESS OF OUR COMMAND TABLE
000352' 062701  000240                  ADD     #TBLD-.,R1
000356' 006300                          ASL     R0              ;MAKE COMMAND A BYTE OFFSET
000360' 060001                          ADD     R0,R1           ;USE IT TO INDEX INTO COMMAND TABLE
000362' 061101                          ADD     (R1),R1         ;R1 NOW HAS COMMAND ROUTINE ADDRESS
000364' 004711                          JSR     PC,(R1)         ;EXECUTE AS COMMANDED FROM HOST
000366' 103404                          BCS     40$             ;ERROR OCCURRED
000370' 112767  000001  000230          MOVB    #INMON,PCSR1    ;INDICATE TO HOST WE ARE BACK IN...
000376' 000403                          BR      45$             ;MICROMONITR
000400' 112767  000003  000220  40$:    MOVB    #INERR,PCSR1    ;INDICATE TO HOST ERROR OCCURRED
000406' 016737  000214  021020  45$:    MOV     PCSR1,@#IPCSR1
000414' 012737  004000  021000          MOV     #DNI,@#IPCSR0   ;TELL HOST THIS MICROTEST FINISHED
000422' 005067  000174          50$:    CLR     FLG5            ;RESET FLAG WORD
000426' 000716                          BR      15$             ;GO WAIT FOR ANOTHER COMMAND

000430' 005000                  60$:    CLR     R0              ;FAKE SUCCESSFUL SELF TEST RESULTS
000432' 000137  040006                  JMP     @#40006         ;START OPERATIONAL MICROCODE

000436' 052767  000001  000156  CSRWRT: BIS     #CSRFLG,FLG5    ;INDICATE A CSR WRITE INTERRUPT OCCURED
000444' 000002                          RTI

000446' 052767  000002  000146  ERRINT: BIS     #ERRFLG,FLG5    ;INDICATE A UNEXPECTED INTERRUPT OCCURED
000454' 012737  020000  021000          MOV     #UNIERR,@#IPCSR0 ;TELL HOST AN UNEXPECTED INTERRUPT
                                                                ;HAPPENED
000462' 000777                          BR      .               ;JUST SIT HERE AND SPIN WHEELS
                                                                ;COUNT ON HOST TO TIMEOUT


000464' 005267  000134          TIMINT: INC     SANTIM          ;COUNT TICKS AS THEY OCCUR
000470' 000002                          RTI

000472' 013767  021002  000140  DMAINT: MOV     @#DMACSR,DMDONE ;GET DMA STATUS
000500' 032767  040000  000132          BIT     #BIT14,DMDONE   ;DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
000506' 001404                          BEQ     10$             ;NO
000510' 012737  040000  021000          MOV     #NXMERR,@#IPCSR0 ;YES, TELL HOST A NON-EXISTANT MEMORY
                                                                ;LOCATION WAS ADDRESSED
000516' 000407                          BR      20$
000520' 032767  100000  000112  10$:    BIT     #BIT15,DMDONE   ;DID A NPR TIMEOUT OCCUR?
```

```
000526' 001407                           BEQ     30$                     ;NO
000530' 012737  100000  021000           MOV     #NPRERR,@#IPCSR0        ;TELL HOST NPR TIMEOUT HAPPENED
000536' 012737  100000  021002  20$:     MOV     #BIT15,@#DMACSR         ;CLEAR THE INTERRUPT IN THE DMA ENGINE
000544' 000777                           BR      .                       ;SIT HERE AND SPIN WHEELS
000546' 000002                  30$:     RTI

000550' 052767  000004  000044  PARINT:  BIS     #PARFLG,FLG5            ;SET PARITY ERROR OCCURRED
000556' 012737  010000  021000           MOV     #PARERR,@#IPCSR0        ;TELL HOST A LINK MEMORY PARITY ERROR
                                                                         ;OCCURRED
000564' 000777                           BR      .                       ;SIT HERE AND SPIN WHEELS

000566' 005737  021044          RCVINT:  TST     @#LRBUF                 ;READ BUFFER DONE REGISTER...
                                                                         ;WHICH CLEARS THE INTERRUPT
000572' 052767  000100  000022           BIS     #RCVFLG,FLG5            ;SET RECEIVER INTERRUPT OCCURRED
000600' 000002                           RTI

000602' 052767  000040  000012  TRNINT:  BIS     #TRNFLG,FLG5            ;SET TRANSMITTER INTERRUPT OCCURRED
000610' 000002                           RTI

000612' 000034                  TBLD:    .WORD   MICE1-.                 ; STATION ADDRESS PATTERN
000614' 000614                           .WORD   MICE2-.                 ; STATION ADDRESS REJECTION
000616' 001064                           .WORD   MICE3-.                 ; STATION ADDRESS POSITION
000620' 001620                           .WORD   MICE4-.                 ; MULTICAST ADDRESS TEST

000622' 000000                  FLG5:    .WORD   0                       ;FLAG WORD
000624' 000000                  SANTIM:  .WORD   0                       ;COUNT FOR SANITY TIMER
000626' 000000                  PCSR1:   .WORD   0                       ;COPY OF WHAT GOES TO PCSR1
000630' 000000  000000          IPCSR2:  .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCSR2
000634' 000000  000000          PCBADR:  .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCB
000640' 000000                  DMDONE:  .WORD   0
000642' 000000                  RBUFP:   .WORD   0                       ;POINTER TO RECIEVE BUFFER
000644' 000000                  TBUFP:   .WORD   0                       ;POINTER TO XMIT BUFFER
```

```
                                        ;
                                        .SBTTL  STATION ADDRESS PATTERN TEST
                                        ;
                                        ; MICROCODE FOR STATION ADDRESS PATTERN TEST. GETS A ''PATTERN'' TO BE
                                        ; USED AS A STATION ADDRESS FOR TESTING THE STATION ADDRESS RAM AND
                                        ; DETECTION CIRCUITRY. FILLS THE STATION ADDRESS RAM. LOOPS A DATAGRAM
                                        ; WITH A STATION ADDRESS IDENTICAL TO THE ADDRESS FILLING STATION
                                        ; ADDRESS RAM. IF ALL IS WORKING, DATAGRAM SHOULD BE RECEIVED OK.
                                        ;
                                        ;
                                        ; ***********************************
                                        ; ***** TELL HIM WE ARE TESTING *****
                                        ; ***********************************
                                        ;
000646' 112767  000002  177752  MICE1:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
000654' 016737  177746  021020          MOV     PCSR1,@#IPCSR1
                                        ;
                                        ;
                                        ; ****************************************
                                        ; ***** PICK UP HOST ADDRESS OF PCBB *****
                                        ; ****************************************
                                        ;
000662' 016737  177742  021010          MOV     IPCSR2,@#MDMA0          ; PICK UP ADDRESS OF PCBB
000670' 016737  177736  021012          MOV     IPCSR2+2,@#MDMA1
000676' 013700  021014                  MOV     @#MDMAR0,R0             ; R0=CONTENTS OF PCSR2
000702' 013701  021014                  MOV     @#MDMAR0,R1             ; R1=CONTENTS OF PCSR3
                                        ;
000706' 010037  021010                  MOV     R0,@#MDMA0              ; POINT TO HOST PCBB
000712' 010137  021012                  MOV     R1,@#MDMA1
                                        ;
000716' 013702  021014                  MOV     @#MDMAR0,R2             ; R2 NOW HOLDS LS ADDRESS WORD
                                        ;
000722' 062700  000002                  ADD     #2,R0                   ; INDEX TO NEXT HOST WORD
000726' 005501                          ADC     R1
                                        ;
000730' 013703  021014                  MOV     @#MDMAR0,R3             ; R3 NOW HOLDS MIDDLE PATTERN
                                        ;
000734' 062700  000002                  ADD     #2,R0                   ; INDEX TO NEXT HOST WORD
000740' 005501                          ADC     R1
                                        ;
000742' 013704  021014                  MOV     @#MDMAR0,R4             ; R4 NOW HOLDS MS PATTERN WORD
                                        ;
000746' 010446                          MOV     R4,-(SP)                ; SAVE THEM FOR LATER
000750' 010346                          MOV     R3,-(SP)
000752' 010246                          MOV     R2,-(SP)
                                        ;
                                        ;
                                        ; *********************************************************************
                                        ; ***** MAKE A STATION ADDRESS FILE FROM THE PASSED PATTERN *****
                                        ; *********************************************************************
                                        ;
000754' 010700                          MOV     PC,R0                   ; POINT AT SA FILE
000756' 062700  000342                  ADD     #AFILE-.,R0             ;
                                        ;
000762' 012705  000014                  MOV     #12.,R5                 ; NEED TOTAL OF 12 ENTRIES
000766' 010220                  10$:    MOV     R2,(R0)+                ; LS WORD OF PATTERN
```

```
000770' 010320                          MOV     R3,(R0)+               ; MIDDLE WORD OF PATTERN
000772' 010420                          MOV     R4,(R0)+               ; MS WORD OF PATTERN
000774' 005305                          DEC     R5                     ; END OF LOOP
000776' 001373                          BNE     10$
                                ;
                                ;
                                ; ****************************************
                                ; ***** LOAD THE STATION ADDRESS RAM *****
                                ; ****************************************
                                ;
001000' 012737 000200 177776            MOV     #MODE,@#CMDREG         ; SET LOOPBACK BEFORE LOADING
001006' 012737 000004 177774            MOV     #LOOP,@#MODREG
                                ;
001014' 010701                          MOV     PC,R1                  ; FORM POS/IND POINTER
001016' 062701 000302                   ADD     #AFILE-.,R1
                                ;
001022' 005000                          CLR     R0                     ; THIS WILL CLEAR HIGH BITS
                                ;
001024' 012746 000120                   MOV     #ARAM+20,-(SP)         ; SA RAM STARTS AT +20 LOCATIONS
                                ;
001030' 012702 000003                   MOV     #3,R2                  ; 3 WORDS PER ADDRESS/PATTERN
001034' 012704 000020           20$:    MOV     #16.,R4                ; SIXTEEN BITS PER WORD
001040' 012705 000014           30$:    MOV     #12.,R5                ; 12 POSITIONS IN SA RAM
                                ;
001044' 010103                          MOV     R1,R3                  ; COPY THE POINTER
                                ;
001046' 006013                  40$:    ROR     (R3)                   ; GET LSB OF ALL 12 ADDRESSES
001050' 006100                          ROL     R0                     ; R0 WILL HOLD ORTHOGONAL WORD
001052' 062703 000006                   ADD     #6,R3                  ; 6 BYTES PER ADDRES/PATTERN
001056' 077505                          SOB     R5,40$                 ; GO TILL DONE
                                ;
001060' 011637 177776                   MOV     (SP),@#CMDREG          ; SET MODE TO WRITE SA RAM
001064' 010037 177774                   MOV     R0,@#ADRREG            ; ORTHOGONAL WORD TO SA RAM
001070' 005216                          INC     (SP)                   ; BUMP STATION ADDRESS
001072' 077416                          SOB     R4,30$                 ; DO ANOTHER ONE
                                ;
001074' 062701 000002                   ADD     #2,R1                  ; ADVANCE TO NEXT WORD
001100' 077223                          SOB     R2,20$                 ; LOOP TILL DONE
                                ;
001102' 012600                          MOV     (SP)+,R0               ; R0 IS A JUNK REGISTER
                                ;
                                ;
                                ; **************************************************
                                ; ***** FILL THE RECEIVE BUFFER WITH BACKGROUND *****
                                ; **************************************************
                                ;
001104' 012700 100000                   MOV     #LINADR,R0             ; RECEIVE BUFFER STARTS HERE
001110' 010067 177526                   MOV     R0,RBUFP
001114' 005020                  50$:    CLR     (R0)+                  ; FILL RECEIVE BUFFER WITH ZEROS
001116' 020027 104000                   CMP     R0,#LINADR+SIZ1K       ; FILL THE BUFFER
001122' 103774                          BLO     50$                    ; FILL ENTIRE BUFFER
                                ;
                                ;
                                ; ************************************************
                                ; ***** FILL TRANSMIT BUFFER WITH TEST PATTERN *****
                                ; ************************************************
```

```
                                        ;
001124' 012703  000377                          MOV     #0377,R3                ; WORST CASE FOR CLOCKING
001130' 010067  177510                          MOV     R0,TBUFP
001134' 010320                          60$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
001136' 020027  110000                          CMP     R0,#LINADR+SIZ2K        ; STOP AT TOP
001142' 103774                                  BLO     60$
                                        ;
                                        ;
                                        ;       ***********************************************************
                                        ;       ***** SET UP LINK FOR DATAGRAM LOOPBACK OPERATION *****
                                        ;       ***********************************************************
                                        ;
001144' 012737  100020  177776                  MOV     #ENABLE+20,@#CMDREG     ; LEAVE 20 IN COMMAND REGISTER
                                        ;
001152' 016701  177466                          MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
001156' 005021                                  CLR     (R1)+                   ; CLEAR OUT STATUS WORD
001160' 012721  000100                          MOV     #MINBC,(R1)+            ; SET BYTE COUNT TO MIN ALLOWED
                                        ;
001164' 012621                                  MOV     (SP)+,(R1)+             ; GET BACK HIGH ADDRESS PATTERN
001166' 012621                                  MOV     (SP)+,(R1)+
001170' 012621                                  MOV     (SP)+,(R1)+             ; GET BACK LOW ADDRESS PATTERN
                                        ;
001172' 005037  021034                          CLR     @#CLRFIF                ; CLEAR THE FIFO
001176' 005067  177420                          CLR     FLGS                    ; CLEAR INTERRUPT FLAG
001202' 016737  177434  021032                  MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
001210' 016737  177430  021030                  MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                        ;
001216' 106427  000140                          MTPS    #PRI03                  ; ALLOW XMIT AND REC TO INTER
001222' 112767  000003  177376                  MOVB    #INERR,PCSR1            ; TELL HOST IN CASE OF FAILURE
001230' 016737  177372  021020                  MOV     PCSR1,@#IPCSR1
                                        ;
001236' 016701  177402                          MOV     TBUFP,R1                ; WAIT FOR MATCH BIT FIRST
001242' 011102                          65$:    MOV     (R1),R2
001244' 032702  020000                          BIT     #MTCH,R2
001250' 001774                                  BEQ     65$
                                        ;
001252' 112767  000002  177346                  MOVB    #INTST,PCSR1            ; TELL HOST WE GOT BY IT
001260' 016737  177342  021020                  MOV     PCSR1,@#IPCSR1
                                        ;
001266' 032767  000100  177326          70$:    BIT     #RCVFLG,FLGS            ; WAIT FOR RECEIVE INTER
001274' 001774                                  BEQ     70$
                                        ;
001276' 032767  000040  177316          80$:    BIT     #TRNFLG,FLGS            ; WAIT FOR XMIT INT TOO!
001304' 001774                                  BEQ     80$
                                        ;
001306' 112767  000001  177313                  MOVB    #1,PCSR1+1              ; TELL HOST WE ARE DONE
001314' 000241                                  CLC
001316' 000207                                  RTS     PC
                                        ;
                                        ;
001320' 000000  000000  000000          AFILE:  .WORD   0,0,0           ; WORD #1
001326' 000000  000000  000000                  .WORD   0,0,0
001334' 000000  000000  000000                  .WORD   0,0,0
001342' 000000  000000  000000                  .WORD   0,0,0
001350' 000000  000000  000000                  .WORD   0,0,0
001356' 000000  000000  000000                  .WORD   0,0,0
```

```
  445  001364' 000000  000000  000000            .WORD   0,0,0
  446  001372' 000000  000000  000000            .WORD   0,0,0
  447  001400' 000000  000000  000000            .WORD   0,0,0
  448  001406' 000000  000000  000000            .WORD   0,0,0
  449  001414' 000000  000000  000000            .WORD   0,0,0
  450  001422' 000000  000000  000000            .WORD   0,0,0
```

```
                                        ;
                                        .SBTTL  STATION ADDRESS REJECTION TEST
                                        ;
                                        ;       THIS TEST WILL VERIFY THAT THE STATION ADDRESS LOGIC WILL NOT
                                        ;       ACCEPT A STATION ADDRESS IF THAT STATION ADDRESS IS NOT PRESENT
                                        ;       IN THE STATION ADDRESS RAM. STATION ADDRESS RAM WILL BE FILLED
                                        ;       WITH AN ADDRESS. DIFFERENT ADDRESS WILL BE PUT IN DATAGRAM FIELD
                                        ;       AND LOOPED BACK. LINK SHOULD NOT RECOGNIZE THE DATAGRAM.
                                        ;
                                        ; ************************************
                                        ; ***** TELL HIM WE ARE TESTING *****
                                        ; ************************************
                                        ;
001430' 112767  000002  177170  MICE2:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
001436' 016737  177164  021020          MOV     PCSR1,@#IPCSR1
                                        ;
001444' 005067  177154                  CLR     SANTIM                  ; CLEAR FLAG FOR TIMER
                                        ;
                                        ;
                                        ; ****************************************************************
                                        ; ***** FILL STATION ADDRESS RAM WITH KNOWN (PHONEY) ADDRESS *****
                                        ; ****************************************************************
                                        ;
001450' 012737  000200  177776          MOV     #MODE,@#CMDREG          ; SET LOOP TO LOAD SA RAM
001456' 012737  000004  177774          MOV     #LOOP,@#MODREG
                                        ;
001464' 012704  000060                  MOV     #48.,R4                 ; COUNTER
001470' 005000                          CLR     R0
                                        ;
001472' 012701  000120                  MOV     #ARAM+20,R1             ; STATION ADDRESS STARTS AT +20
001476' 010137  177776          10$:    MOV     R1,@#CMDREG             ; SELECT STATION ADDRESS RAM
001502' 010037  177774                  MOV     R0,@#ADRREG             ; PARK IN RAM
001506' 005201                          INC     R1                      ; NEXT LOCATION
001510' 077406                          SOB     R4,10$                  ; DO THEM ALL
                                        ;
                                        ;
                                        ; ****************************************************
                                        ; ***** FILL THE RECEIVE BUFFER WITH BACKGROUND *****
                                        ; ****************************************************
                                        ;
001512' 012700  100000                  MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
001516' 010067  177120                  MOV     R0,RBUFP
001522' 005020                  20$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
001524' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ; FILL THE BUFFER
001530' 103774                          BLO     20$                     ; FILL ENTIRE BUFFER
                                        ;
                                        ;
                                        ; ***************************************************
                                        ; ***** FILL TRANSMIT BUFFER WITH TEST PATTERN *****
                                        ; ***************************************************
                                        ;
001532' 012703  000377                  MOV     #0377,R3                ; WORST CASE FOR CLOCKING
001536' 010067  177102                  MOV     R0,TBUFP
001542' 010320                  30$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
001544' 020027  110000                  CMP     R0,#LINADR+SIZ2K        ; STOP AT TOP
001550' 103774                          BLO     30$
```

```
                                      ;
                                      ; ***************************************************
                                      ; ***** SET UP LINK FOR DATAGRAM LOOPBACK OPERATION *****
                                      ; ***************************************************
                                      ;
001552' 012737  100020  177776                MOV     #ENABLE+20,@#CMDREG     ; POINTER TO FRONT OF RAM
                                      ;
001560' 016701  177060                        MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
001564' 005021                                CLR     (R1)+                   ; CLEAR OUT STATUS WORD
001566' 012721  000100                        MOV     #MINBC,(R1)+            ; SET BYTE COUNT TO MIN ALLOWED
                                      ;
                                      ;
                                      ; *****************************************
                                      ; ***** GET PHYSICAL ADDRESS INTO DATAGRAM *****
                                      ; *****************************************
                                      ;
001572' 012700  177777                        MOV     #177777,R0              ; && STAND-IN FOR PHYSICAL ADDR
001576' 010021                                MOV     R0,(R1)+
001600' 010021                                MOV     R0,(R1)+
001602' 010021                                MOV     R0,(R1)+
                                      ;
                                      ;
001604' 005037  021034                        CLR     @#CLRFIF                ; CLEAR THE FIFO
001610' 005067  177006                        CLR     FLG5                    ; CLEAR INTERRUPT FLAG
001614' 016737  177022  021032                MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
001622' 016737  177016  021030                MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                      ;
001630' 106427  000140                        MTPS    #PRI03                  ; ALLOW XMIT AND REC TO INTER
001634' 026727  176764  000002  40$:          CMP     SANTIM,#2               ; WAIT FOR 2 SECONDS
001642' 002012                                BGE     50$                     ; EXIT NORMALLY IF TIMER DONE
001644' 032767  000100  176750                BIT     #RCVFLG,FLG5            ; WAIT FOR RECEIVER INTERRUPT
001652' 001770                                BEQ     40$                     ; IF NONE, WE'RE OK
                                      ;
                                      ;
                                      ; ****************************
                                      ; ***** ERROR FALLTHROUGH *****
                                      ; ****************************
                                      ;
001654' 032767  000040  176740  45$:          BIT     #TRNFLG,FLG5            ; WAIT FOR XMIT INT TOO!
001662' 001774                                BEQ     45$
                                      ;
001664' 000261                                SEC                             ;INDICATE ERROR
001666' 000401                                BR      60$
                                      ;
                                      ;
                                      ; ****************************
                                      ; ***** EXIT FOR NO ERROR *****
                                      ; ****************************
                                      ;
001670' 000241                        50$:    CLC                             ;INDICATE SUCESS
001672' 112767  000002  176727        60$:    MOVB    #2,PCSR1+1              ; TELL HOST WE ARE DONE
001700' 000207                                RTS     PC
                                      ;
                                      ;
```

```
                                          ;
                                          .SBTTL  STATION ADDRESS RAM POSITION TEST
                                          ;
                                          ;       THIS TEST WILL CHECK THAT A STATION ADDRESS IS RECOGNIZED
                                          ;       REGARDLESS OF WHICH OF THE 12 RAM POSITIONS THE ADDRESS
                                          ;       RESIDES. THE PHYSICAL ADDRESS WILL BE WRITTEN TO EACH OF THE 12
                                          ;       STATION ADDRESS RAM POSITIONS WITH THE REST OF THE POSITIONS
                                          ;       FILLED WITH KNOWN DATA. A DATAGRAM WITH THE PHSYICAL ADDRESS
                                          ;       WILL BE LOOPED AROUND. THE TEST WILL VERIFY THE DATAGRAM IS
                                          ;       RECEIVED.
                                          ;
                                          ; ***************************************
                                          ; ***** TELL HIM WE ARE TESTING *****
                                          ; ***************************************
                                          ;
001702' 112767  000002  176716  MICE3:    MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
001710' 016737  176712  021020            MOV     PCSR1,@#IPCSR1
                                          ;
001716' 005067  176702                    CLR     SANTIM                  ; CLEAR THE FLAG
                                          ;
                                          ;
                                          ; *****************************************
                                          ; ***** PICK UP HOST ADDRESS OF PCBB *****
                                          ; *****************************************
                                          ;
001722' 016737  176702  021010            MOV     IPCSR2,@#MDMA0          ; PICK UP ADDRESS OF PCBB
001730' 016737  176676  021012            MOV     IPCSR2+2,@#MDMA1
001736' 013700  021014                    MOV     @#MDMAR0,R0             ; R0=CONTENTS OF PCSR2
001742' 013701  021014                    MOV     @#MDMAR0,R1             ; R1=CONTENTS OF PCSR3
                                          ;
001746' 010037  021010                    MOV     R0,@#MDMA0              ; POINT TO HOST PCBB
001752' 010137  021012                    MOV     R1,@#MDMA1
                                          ;
001756' 013702  021014                    MOV     @#MDMAR0,R2             ; R2 NOW HOLDS SA RAM POSITION
                                          ;
                                          ;
                                          ; ********************************************************
                                          ; ***** FILL THE BUFFER WITH PHONEY STATION ADDRESSES *****
                                          ; ********************************************************
                                          ;
001762' 010700                            MOV     PC,R0                   ; CALCULATE POS/IND ADDRESS
001764' 062700  000344                    ADD     #BFILE-.,R0
                                          ;
001770' 012705  000044                    MOV     #36.,R5                 ; THERE ARE 12 ADDRESSES
001774' 005020                    10$:    CLR     (R0)+                   ; FILL WITH ZERO (ADDRESS)
001776' 005305                            DEC     R5
002000' 001375                            BNE     10$
                                          ;
                                          ;
                                          ; ************************************************
                                          ; ***** PUT PHYSICAL ADDRESS IN RAM POSITION *****
                                          ; ************************************************
                                          ;
002002' 010700                            MOV     PC,R0                   ; CALCULATE POS/IND ADDRESS
002004' 062700  000324                    ADD     #BFILE-.,R0
                                          ;
```

```
                                        ;
002010' 005302                          20$:    DEC     R2                      ; PARAMTER IS NEVER > 1
002012' 001403                                  BEQ     30$                     ; GO TO WORK
002014' 062700  000006                          ADD     #6,R0
002020' 000773                                  BR      20$                     ; GO ROUND
                                        ;
002022' 012701  177777                  30$:    MOV     #177777,R1              ; && STAND-IN FOR PHSICAL ADDR
002026' 010120                                  MOV     R1,(R0)+
002030' 010120                                  MOV     R1,(R0)+
002032' 010120                                  MOV     R1,(R0)+
                                        ;
                                        ;
                                        ; ***************************************
                                        ; ***** LOAD THE STATION ADDRESS RAM *****
                                        ; ***************************************
                                        ;
002034' 012737  000200  177776                  MOV     #MODE,@#CMDREG          ; SET LOOP TO LOAD SA RAM
002042' 012737  000004  177774                  MOV     #LOOP,@#MODREG
                                        ;
002050' 010701                                  MOV     PC,R1                   ; FORM A POS/IND ADDRES
002052' 062701  000256                          ADD     #BFILE-.,R1             ; ADD THE OFFSET
                                        ;
002056' 005000                                  CLR     R0                      ; THIS CLEARS HIGH BITS
                                        ;
002060' 012746  000120                          MOV     #ARAM+20,-(SP)          ; FOR MODE REG/NEED A REGISTER
                                        ;
002064' 012702  000003                          MOV     #3,R2                   ; 3 WORDS PER ADDRESS/PATTERN
002070' 012704  000020                  40$:    MOV     #16.,R4                 ; SIXTEEN BITS PER WORD
002074' 012705  000014                  50$:    MOV     #12.,R5                 ; 12 POSITIONS IN SA RAM
                                        ;
002100' 010103                                  MOV     R1,R3                   ; COPY THE POINTER
                                        ;
002102' 006013                          60$:    ROR     (R3)                    ; GET LSB OF ALL 12 ADDRESSES
002104' 006100                                  ROL     R0                      ; R0 WILL HOLD ORTHOGONAL WORD
002106' 062703  000006                          ADD     #6,R3                   ; 6 BYTES PER ADDRES/PATTERN
002112' 077505                                  SOB     R5,60$                  ; GO TILL DONE
                                        ;
002114' 011637  177776                          MOV     (SP),@#CMDREG           ; SET MODE TO WRITE SA RAM
002120' 010037  177774                          MOV     R0,@#ADRREG             ; ORTHOGONAL WORD TO SA RAM
002124' 005216                                  INC     (SP)                    ; BUMP STATION ADDRESS
002126' 077416                                  SOB     R4,50$                  ; DO ANOTHER ONE
                                        ;
002130' 062701  000002                          ADD     #2,R1                   ; ADVANCE TO NEXT WORD
002134' 077223                                  SOB     R2,40$                  ; LOOP TILL DONE
                                        ;
002136' 012600                                  MOV     (SP)+,R0                ; POP TO BYTE BUCKET
                                        ;
                                        ;
                                        ; ************************************************
                                        ; ***** FILL THE RECEIVE BUFFER WITH BACKGROUND *****
                                        ; ************************************************
                                        ;
002140' 012700  100000                          MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
002144' 010067  176472                          MOV     R0,RBUFP
002150' 005020                          70$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
002152' 020027  104000                          CMP     R0,#LINADR+SIZ1K        ; FILL THE BUFFER
```

```
002156' 103774                          BLO     70$                     ; FILL ENTIRE BUFFER
                                ;
                                ;
                                ; ****************************************************
                                ; ***** FILL TRANSMIT BUFFER WITH TEST PATTERN *****
                                ; ****************************************************
                                ;
002160' 012703  000377                  MOV     #0377,R3                ; WORST CASE FOR CLOCKING
002164' 010067  176454                  MOV     R0,TBUFP
002170' 010320                  80$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
002172' 020027  110000                  CMP     R0,#LINADR+SIZ2K        ; STOP AT TOP
002176' 103774                          BLO     80$
                                ;
                                ;
                                ; **********************************************************
                                ; ***** SET UP LINK FOR DATAGRAM LOOPBACK OPERATION *****
                                ; **********************************************************
                                ;
002200' 012737  100020  177776          MOV     #ENABLE+20,@#CMDREG     ; ENABLE LINK
                                ;
002206' 016701  176432                  MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
002212' 005021                          CLR     (R1)+                   ; CLEAR OUT STATUS WORD
002214' 012721  000100                  MOV     #MINBC,(R1)+            ; SET BYTE COUNT TO MIN ALLOWED
                                ;
002220' 012700  177777                  MOV     #177777,R0              ; && STAND IN PHYSICAL ADDRESS
002224' 010021                          MOV     R0,(R1)+
002226' 010021                          MOV     R0,(R1)+
002230' 010021                          MOV     R0,(R1)+
                                ;
002232' 005037  021034                  CLR     @#CLRFIF                ; CLEAR THE FIFO
002236' 005067  176360                  CLR     FLG5                    ; CLEAR INTERRUPT FLAG
002242' 016737  176374  021032          MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
002250' 016737  176370  021030          MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                ;
002256' 106427  000140                  MTPS    #PRI03                  ; ALLOW XMIT AND REC TO INTER
002262' 026727  176336  000002  90$:    CMP     SANTIM,#2               ; WAIT FOR COUNTS TO ACCUMULATE
002270' 002015                          BGE     120$
002272' 032767  000100  176322          BIT     #RCVFLG,FLG5            ; THIS SETS IF DATAGRAM LOOPS
002300' 001770                          BEQ     90$
                                ;
002302' 032767  000040  176312  100$:   BIT     #TRNFLG,FLG5            ; WAIT FOR XMIT TOO
002310' 001774                          BEQ     100$
                                ;
002312' 000241                          CLC
002314' 112767  000003  176305  110$:   MOVB    #3,PCSR1+1              ; TELL HIM WHAT TEST WE'RE IN
002322' 000207                          RTS     PC
                                ;
                                ;
002324' 000261                  120$:   SEC                             ;INDICATE ERROR
002326' 000772                          BR      110$                    ; GO BACK TO MICROMONITOR
                                ;
                                ;
002330' 000000  000000  000000  BFILE:: .WORD   0,0,0
002336' 000000  000000  000000          .WORD   0,0,0
002344' 000000  000000  000000          .WORD   0,0,0
002352' 000000  000000  000000          .WORD   0,0,0
```

```
002360' 000000  000000  000000          .WORD   0,0,0
002366' 000000  000000  000000          .WORD   0,0,0
002374' 000000  000000  000000          .WORD   0,0,0
002402' 000000  000000  000000          .WORD   0,0,0
002410' 000000  000000  000000          .WORD   0,0,0
002416' 000000  000000  000000          .WORD   0,0,0
002424' 000000  000000  000000          .WORD   0,0,0
002432' 000000  000000  000000          .WORD   0,0,0
```

```
                                          ;
                                          .SBTTL  MULTICAST ADDRESS TEST
                                          ;
                                          ; ***********************************
                                          ; ***** TELL HIM WE ARE TESTING *****
                                          ; ***********************************
                                          ;
002440' 112767  000002  176160   MICE4:   MOVB    #INTST,PCSR1             ; TELL HOST WE ARE TESTING
002446' 016737  176154  021020            MOV     PCSR1,@#IPCSR1
                                          ;
002454' 005067  176144                    CLR     SANTIM                   ; CLEAR FLAG FOR TIMER
                                          ;
                                          ;
                                          ; *****************************************************************
                                          ; ***** FILL STATION ADDRESS RAM WITH KNOWN (PHONEY) ADDRESS *****
                                          ; *****************************************************************
                                          ;
002460' 012737  000200  177776            MOV     #MODE,@#CMDREG           ; SET LOOP TO LOAD SA RAM
002466' 012737  000004  177774            MOV     #LOOP,@#MODREG
                                          ;
002474' 012704  000060                    MOV     #48.,R4                  ; COUNTER
002500' 005000                            CLR     R0
                                          ;
002502' 012701  000120                    MOV     #ARAM+20,R1              ; STATION ADDRESS STARTS AT +20
002506' 010137  177776           10$:     MOV     R1,@#CMDREG              ; SELECT STATION ADDRESS RAM
002512' 010037  177774                    MOV     R0,@#ADRREG              ; PARK IN RAM
002516' 005201                            INC     R1                       ; NEXT LOCATION
002520' 077406                            SOB     R4,10$                   ; DO THEM ALL
                                          ;
                                          ;
                                          ; ***********************************************
                                          ; ***** FILL THE RECEIVE BUFFER WITH BACKGROUND *****
                                          ; ***********************************************
                                          ;
002522' 012700  100000                    MOV     #LINADR,R0               ; RECEIVE BUFFER STARTS HERE
002526' 010067  176110                    MOV     R0,RBUFP
002532' 005020                   20$:     CLR     (R0)+                    ; FILL RECEIVE BUFFER WITH ZEROS
002534' 020027  104000                    CMP     R0,#LINADR+SIZ1K         ; FILL THE BUFFER
002540' 103774                            BLO     20$                      ; FILL ENTIRE BUFFER
                                          ;
                                          ;
                                          ; **********************************************
                                          ; ***** FILL TRANSMIT BUFFER WITH TEST PATTERN *****
                                          ; **********************************************
                                          ;
002542' 012703  000377                    MOV     #0377,R3                 ; WORST CASE FOR CLOCKING
002546' 010067  176072                    MOV     R0,TBUFP
002552' 010320                   30$:     MOV     R3,(R0)+                 ; FILL XMIT BUFFER WITH PATTERN
002554' 020027  110000                    CMP     R0,#LINADR+SIZ2K         ; STOP AT TOP
002560' 103774                            BLO     30$
                                          ;
                                          ;
                                          ; ***************************************************
                                          ; ***** SET UP LINK FOR DATAGRAM LOOPBACK OPERATION *****
                                          ; ***************************************************
                                          ;
                                          ;
```

```
002562' 012737  000200  177776          MOV     #MODE,@#CMDREG          ; ENABLE LINK, SELECT MODE
002570' 012737  040004  177774          MOV     #ENAL!LOOP,@#MODREG     ; ENABLE LOOPBACK
002576' 012737  100020  177776          MOV     #ENABLE+20,@#CMDREG     ; POINTER TO FRONT OF RAM
                                ;
002604' 016701  176034                  MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
002610' 005021                          CLR     (R1)+                   ; CLEAR OUT STATUS WORD
002612' 012721  000100                  MOV     #MINBC,(R1)+            ; SET BYTE COUNT TO MIN ALLOWED
                                ;
                                ;
                                ; ***********************************************
                                ; ***** GET PHYSICAL ADDRESS INTO DATAGRAM *****
                                ; ***********************************************
                                ;
002616' 012700  177777                  MOV     #177777,R0              ; && STAND-IN FOR PHYSICAL ADDR
002622' 010011                          MOV     R0,(R1)
002624' 052721  000001                  BIS     #01,(R1)+               ; SET MULTICAST ADDRESS BIT
002630' 010021                          MOV     R0,(R1)+
002632' 010021                          MOV     R0,(R1)+
                                ;
                                ;
002634' 005037  021034                  CLR     @#CLRFIF                ; CLEAR THE FIFO
002640' 005067  175756                  CLR     FLG5                    ; CLEAR INTERRUPT FLAG
002644' 016737  175772  021032          MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
002652' 016737  175766  021030          MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                ;
002660' 106427  000140                  MTPS    #PRI03                  ; ALLOW XMIT AND REC TO INTER
002664' 026727  175734  000002  40$:    CMP     SANTIM,#2               ; && LEAVE IT 2 SECONDS
002672' 002015                          BGE     60$                     ; EXIT IF TIMER DONE
002674' 032767  000100  175720          BIT     #RCVFLG,FLG5            ; WAIT FOR RECEIVER INTERRUPT
002702' 001770                          BEQ     40$                     ;
                                ;
002704' 032767  000040  175710  45$:    BIT     #TRNFLG,FLG5            ; WAIT FOR XMIT INT TOO!
002712' 001774                          BEQ     45$
                                ;
002714' 000241                          CLC                             ;INDICATE SUCCESS
002716' 112767  000004  175703  50$:    MOVB    #4,PCSR1+1              ; TELL HOST WE ARE DONE
002724' 000207                          RTS     PC
                                ;
                                ;
002726' 000261                  60$:    SEC                             ;INDICATE ERROR
002730' 000772                          BR      50$                     ; RETURN TO MICROMONITOR
                                ;
                                ;
```

```
                                   ;
     002732' 002734                MICESZ::MICESZ-MICROE+2
                                   ;
             000001                .END
```

```
ACLO   = 002000             87#
ADRERR= 000001             130#
ADRREG= 177774              34#    365*    482*    656*    763*
AFILE    001320R    002    329     347     439#
ARAM   = 000100             80#    351     480     642     701
BFILE    002330RG   002    603     616     638     726#
BIT0   = 000001             54#    103
BIT1   = 000002             53#    104
BIT10  = 002000             44#     56      73      87
BIT11  = 004000             43#     72      86
BIT12  = 010000             42#     56      71     113
BIT13  = 020000             41#     70     112     114
BIT14  = 040000             40#     69      85     111     238
BIT15  = 100000             39#     68      78      84     110     243     246
BIT2   = 000004             52#     91     105
BIT3   = 000010             51#     90     106
BIT4   = 000020             50#     89     107
BIT5   = 000040             49#     88     108
BIT6   = 000100             48#     80     109
BIT7   = 000200             47#     79
BIT8   = 000400             46#     56      74
BIT9   = 001000             45#
CLRFIF= 021034              22#    411*    530*    703*    814*
CMDREG= 177776              35#    136*    343*    364*    401*    474*    481*    513*    634*    655*    692*    755*    762*
                           794*    796*
COLL   = 000020             89#
CSRFLG= 000001             103#    194     225
CSRVEC= 000064              96#    161*    162*
CSRWRT   000436R    002    160     225#
DATERR= 000000             129#
DMACSR= 021002               9#    237     246*
DMAF   = 021022             17#
DMAINT   000472R    002    155     237#
DMAT0  = 021004             10#
DMAT1  = 021006             11#
DMAVEC= 000114              97#    156*    157*
DMAWC  = 021024             18#
DMDONE   000640R    002    237*    238     243     273#
DNI    = 004000             72#    140     218
DRTY   = 000040             88#
DTCR   = 000010             90#
ENABLE= 100000              78#    401     513     692     796
ENAL   = 040000             85#    795
ENCR   = 004000             86#
ERRFLG= 000002             104#    228
ERRINT   000446R    002    142     228#
FATI   = 000400             74#
FLG5     000622R    002    187*    190     194     219*    225*    228*    250*    257*    260*    268#    412*    428     431
                           531*    538     546     704*    711     714     815*    822     825
INERR  = 000003            102#    216     417
INMON  = 000001            100#    138     202     214
INTST  = 000002            101#    291     425     464     577     745
IOADR  = 020000            124#    125
IOSIZ  = 020000            120#    125
IPCSR0= 021000               8#    140*    199     218*    229*    240*    245*    251*
IPCSR1= 021020              16#    139*    202*    217*    292*    418*    426*    465*    578*    746*
```

```
IPCSR2  000630R    002    185*   186*   271#   299    300    587    588
LASFTP= 012400             56#
LFRBUF= 021032             21#   413*   532*   705*   816*
LINADR= 100000            126#   379    382    393    491    494    505    670    673    684    772    775    786
LINSIZ= 077774            122#
LOOP  = 000004             91#   344    475    635    756    795
LRBUF = 021044             26#   255
LTAC  = 021030             20#   414*   533*   706*   817*
MAXBC = 002752            128#
MDMAR0= 021014             14#   301    302    307    312    317    589    590    595
MDMAR1= 021016             15#
MDMAW0= 021026             19#
MDMAW1= 021036             23#
MDMA0 = 021010             12#   299*   304*   587*   592*
MDMA1 = 021012             13#   300*   305*   588*   593*
MDMSR = 021042             25#
MICESZ  002732RG   002    838#
MICE1   000646R    002    263    291#
MICE2   001430R    002    264    464#
MICE3   001702R    002    265    577#
MICE4   002440R    002    266    745#
MICROE  000000RG   002    135#   838
MINBC = 000100            127#   405    517    696    800
MODE  = 000200             79#   343    474    634    755    794
MODREG= 177774             33#   344*   475*   635*   756*   795*
MTCH  = 020000            114#   422
NPRERR= 100000            110#   245
NPRFLG= 000020            107#
NXMERR= 040000            111#   240
NXMFLG= 000010            106#
PARERR= 010000            113#   251
PARFLG= 000004            105#   250
PARINT  000550R    002    150    250#
PARVEC= 000140             98#   151*   152*
PCBADR  000634R    002    272#
PCEI  = 040000             69#
PCSRSW= 021040             24#   179
PCSR1   000626R    002    138*   139    214*   216*   217    270#   291*   292    417*   418    425*   426    434*
                          464*   465    558*   577*   578    718*   745*   746    829*
PHYAD0= 021060             27#
PHYAD1= 021062             28#
PHYAD2= 021064             29#
PHYAD3= 021066             30#
PHYAD4= 021070             31#
PHYAD5= 021072             32#
PRI00 = 000000             64#   188*
PRI01 = 000040             63#
PRI02 = 000100             62#
PRI03 = 000140             61#   416*   535*   708*   819*
PRI04 = 000200             60#   162    172
PRI05 = 000240             59#   167    177
PRI06 = 000300             58#
PRI07 = 000340             57#   135*   145    152    157    193*
PROM  = 100000             84#
RBUFP   000642R    002    274#   380*   413    492*   532    671*   705    773*   816
RCEI  = 002000             73#
```

```
RCVFLG= 000100          109#    257     428     538     711     822
RCVINT  000566R   002   175     255#
RCVVEC= 000120           94#    176*    177*
ROMADR= 040000          125#    126
ROMSIZ= 040000          121#    126
RXI   = 020000           70#
SANTIM  000624R   002   234*    269#    467*    536     580*    709     748*    820
SANVEC= 000134           95#    166*    167*
SERI  = 100000           68#
SIZ1K = 004000          115#    116     382     494     673     775
SIZ2K = 010000          116#    117     393     505     684     786
SIZ4K = 020000          117#    118     119     120
SIZ8K = 040000          118#    121     122
STACK = 001000           99#    137
TBLD    000612R   002   208     263#
TBUFP   000644R   002   275#    391*    403     414     420     503*    515     533     682*    694     706     784*    798
                        817
TIMINT  000464R   002   165     234#
TRNFLG= 000040          108#    260     431     546     714     825
TRNINT  000602R   002   170     260#
TRNVEC= 000070           93#    171*    172*
TXI   = 010000           71#
UNIERR= 020000          112#    229
WCSADR= 000000          123#    124
WCSSIZ= 020000          119#    124
.     = 002734R   002   142     150     155     160     165     170     175     196     208     231     247     253     263
                        264     265     266     329     347     603     616     638
```

```
BCOMPL     1#
BERROR     1#
BGNAU      1#
BGNAUT     1#
BGNCLN     1#
BGNDU      1#
BGNHRD     1#
BGNHW      1#
BGNINI     1#
BGNMOD     1#
BGNMSG     1#
BGNPRO     1#
BGNPTA     1#
BGNRPT     1#
BGNSEG     1#
BGNSET     1#
BGNSFT     1#
BGNSRV     1#
BGNSUB     1#
BGNSW      1#
BGNTST     1#
BNCOMP     1#
BNERRO     1#
BREAK      1#
BRESET     1#
CKLOOP     1#
CLOCK      1#
CLOSE      1#
CLRVEC     1#
COMMEN     1#
DELAY      1#
DESCRI     1#
DEVTYP     1#
DISPAT     1#
DISPLA     1#
DOCLN      1#
DODU       1#
DORPT      1#
ENDAU      1#
ENDAUT     1#
ENDCLN     1#
ENDCOM     1#
ENDDU      1#
ENDHRD     1#
ENDHW      1#
ENDINI     1#
ENDMOD     1#
ENDMSG     1#
ENDPRO     1#
ENDPTA     1#
ENDRPT     1#
ENDSEG     1#
ENDSET     1#
ENDSFT     1#
ENDSRV     1#
ENDSUB     1#
```

```
ENDSW       1#
ENDTST      1#
EQUALS      1#
ERRDF       1#
ERRHRD      1#
ERROR       1#
ERRSF       1#
ERRSOF      1#
ERRTBL      1#
ESCAPE      1#
EXIT        1#
FEQUAL      1#
GETBYT      1#
GETPRI      1#
GETWOR      1#
GMANIA      1#
GMANID      1#
GMANIL      1#
GPHARD      1#
GPRMA       1#
GPRMD       1#
GPRML       1#
HEADER      1#
INLOOP      1#
IOSETU      1#
IOSTAR      1#
KT11        1#
LASTAD      1#
MANUAL      1#
MEMORY      1#
MSBYTE      1#
MSCHEC      1#
MSCNTO      1#
MSCOUN      1#
MSDATA      1#
MSDECR      1#
MSDEFA      1#
MSENDE      1#
MSERRI      1#
MSESCA      1#
MSESCS      1#
MSEXCP      1#
MSEXIT      1#
MSEXSE      1#
MSEXTJ      1#
MSGEN       1#
MSGENB      1#
MSGETS      1#
MSGETT      1#
MSGNGB      1#
MSGNIN      1#
MSGNLS      1#
MSGNSU      1#
MSGNTA      1#
MSGNTE      1#
MSHAPT      1#
```

```
MSHNAP      1#
MSINCR      1#
MSIOSE      1#
MSLDRO      1#
MSMASK      1#
MSMCHI      1#
MSMCLO      1#
MSMSK1      1#
MSPOP       1#
MSPRIN      1#
MSPUSH      1#
MSPUT       1#
MSPUT1      1#
MSRADI      1#
MSRBRO      1#
MSRNRO      1#
MSSETS      1#
MSSTAR      1#
MSSVC       1#
MSTLAB      1#
MSTSTL      1#
MSWORD      1#
MSXFER      1#
OPEN        1#
POINTE      1#
PRINTB      1#
PRINTF      1#
PRINTS      1#
PRINTX      1#
READBU      1#
READEF      1#
RFLAGS      1#
SETPRI      1#
SETVEC      1#
SLASH       1#
STARS       1#
SVC         1#
XFER        1#
XFERF       1#
XFERT       1#


. ABS.   000000     000
         000000     001
MICRE    002734     002


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROE.OBJ,MICROE.LST/CR/SOL/NL:TOC=SVC34R.P11,MICROE.MAC
 RUN-TIME: 2 3 .4 SECONDS
 RUN-TIME RATIO: 33/6=5.1
 CORE USED:  31K  (61 PAGES)
```

```
                          .TITLE  MICROF - MICROCODE MODULE F
                          ; && DEDICATED THE CRC CIRCUITRY TO THE RECEIVE SIDE OF THE LINK

        000000'           .CSECT MICRF

                          .SBTTL   REGISTER DEFINITIONS USED BY THE T11

        021000            IPCSR0  =       21000   ;INTERNAL PCSR0 ADDRESS
        021002            DMACSR  =       21002   ;DMA ENGINE CONTROL STATUS REGISTER
        021004            DMAT0   =       21004   ;DMA ENGINE TO ADDRESS REGISTER #0
        021006            DMAT1   =       21006   ;DMA ENGINE TO ADDRESS REGISTER #1
        021010            MDMA0   =       21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
        021012            MDMA1   =       21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
        021014            MDMAR0  =       21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
        021016            MDMAR1  =       21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
        021020            IPCSR1  =       21020   ;INTERNAL PCSR1 ADDRESS
        021022            DMAF    =       21022   ;DMA ENGINE FROM ADDRESS REGISTER
        021024            DMAWC   =       21024   ;DMA ENGINE WORD COUNT REGISTER
        021026            MDMAW0  =       21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
        021030            LTAC    =       21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
        021032            LFRBUF  =       21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
        021034            CLRFIF  =       21034   ;CLEAR FIFO
        021036            MDMAW1  =       21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
        021040            PCSRSW  =       21040   ;SWITCH PACK REGISTER
        021042            MDMSR   =       21042   ;MICROCPU DMA STATUS REGISTER
        021044            LRBUF   =       21044   ;LINK RECIEVE BUFFER COMPLETED
        021060            PHYAD0  =       21060   ;PHYSICAL ADDRESS ROM BYTE 0
        021062            PHYAD1  =       21062   ;PHYSICAL ADDRESS ROM BYTE 1
        021064            PHYAD2  =       21064   ;PHYSCIAL ADDRESS ROM BYTE 2
        021066            PHYAD3  =       21066   ;PHYSICAL ADDRESS ROM BYTE 3
        021070            PHYAD4  =       21070   ;PHYSICAL ADDRESS ROM BYTE 4
        021072            PHYAD5  =       21072   ;PHYSICAL ADDRESS ROM BYTE 5
        177774            MODREG  =       177774  ;LINK MODE REGISTER
        177774            ADRREG  =       177774  ;LINK STATION ADDRESS RAM REGISTER
        177776            CMDREG  =       177776  ;LINK COMMAND REGISTER

                          .SBTTL   OTHER DEFINITIONS USED BY THE MICROCODE

        100000            BIT15   =       100000
        040000            BIT14   =       40000
        020000            BIT13   =       20000
        010000            BIT12   =       10000
        004000            BIT11   =       4000
        002000            BIT10   =       2000
        001000            BIT9    =       1000
        000400            BIT8    =       400
        000200            BIT7    =       200
        000100            BIT6    =       100
        000040            BIT5    =       40
        000020            BIT4    =       20
        000010            BIT3    =       10
        000004            BIT2    =       4
        000002            BIT1    =       2
        000001            BIT0    =       1
                          ;
        012400            LASFTP  =       BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
```

```
000340          PRI07   =       340
000300          PRI06   =       300
000240          PRI05   =       240
000200          PRI04   =       200
000140          PRI03   =       140
000100          PRI02   =       100
000040          PRI01   =       40
000000          PRI00   =       0
                ;
                ;PCSR0 - PORT CONTROL STATUS REGISTER 0
                ;
100000          SERI    =       BIT15
040000          PCEI    =       BIT14
020000          RXI     =       BIT13
010000          TXI     =       BIT12
004000          DNI     =       BIT11
002000          RCEI    =       BIT10
000400          FATI    =       BIT8
                ;
                ;LINK COMMAND REGISTER
                ;
100000          ENABLE  =       BIT15   ;ENABLE LINK MODULE
000200          MODE    =       BIT7    ;ENABLE MODE REGISTER
000100          ARAM    =       BIT6    ;ENABLE STATION ADDRESS RAM
                ;
                ;LINK MODE REGISTER
                ;
100000          PROM    =       BIT15   ;PROMISCUIOUS MODE
040000          ENAL    =       BIT14   ;ENABLE MULTICAST
004000          ENCR    =       BIT11   ;ENABLE COLLISION TEST
002000          ACLO    =       BIT10   ;ENABLE ACLO
000040          DRTY    =       BIT5    ;DISABLE RETRY LOGIC
000020          COLL    =       BIT4    ;SIMULATE A COLLISION
000010          DTCR    =       BIT3    ;DISABLE TRANSMIT CRC LOGIC
000004          LOOP    =       BIT2    ;ENABLE LOOPBACK
000001          HDX     =       BIT0    ;HALF DUPLEX BIT

000070          TRNVEC= 70              ;VECTOR ADDRESS FOR THE TRANSMITTER
000120          RCVVEC= 120             ;VECTOR ADDRESS FOR THE RECEIVER
000134          SANVEC= 134             ;VECTOR ADDRESS FOR THE SANITY TIMER
000064          CSRVEC= 64              ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
000114          DMAVEC= 114             ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
000140          PARVEC= 140             ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
001000          STACK=  1000            ;STACK LOCATION
000001          INMON=  1               ;IN MICROMONITOR STATE
000002          INTST=  2               ;IN A TEST STATE
000003          INERR=  3               ;IN ERROR STATE
000001          CSRFLG= BIT0            ;CSR WRITE INTERRUPT OCCURED
000002          ERRFLG= BIT1            ;UNEXPECTED ERROR OCCURED
000004          PARFLG= BIT2            ;PARITY ERROR OCCURED
000010          NXMFLG= BIT3            ;NON-EXISTANT MEMORY ERROR OCCURRED
000020          NPRFLG= BIT4            ;NPR TIMEOUT OCCURRED
000040          TRNFLG= BIT5            ;TRANSMITTER INTERRUPT OCCURRED
000100          RCVFLG= BIT6            ;RECEIVER INTERRUPT OCCURRED
100000          NPRERR= BIT15           ;PCSR0 FLAG INDICATING NPR ERROR OCCURRED
040000          NXMERR= BIT14           ;PCSR0 FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
```

```
020000      UNIERR= BIT13              ;PCSRO FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
010000      PARERR= BIT12              ;PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
004000      SIZ1K=  4000               ;1K WORDS
010000      SIZ2K=  SIZ1K*2            ;2K WORDS
020000      SIZ4K=  SIZ2K*2            ;4K WORDS
040000      SIZ8K=  SIZ4K*2            ;8K WORDS
020000      WCSSIZ= SIZ4K              ;SIZE OF WRITEABLE CONTROL STORE
020000      IOSIZ=  SIZ4K              ;SIZE OF I/O PAGE
040000      ROMSIZ= SIZ8K              ;SIZE OF ROM
077774      LINSIZ= SIZ8K*2-4          ;SIZE OF LINK MEMORY
000000      WCSADR= 0                  ;BASE ADDRESS OF WCS
020000      IOADR=  WCSADR+WCSSIZ      ;BASE ADDRESS OF I/O PAGE
040000      ROMADR= IOADR+IOSIZ        ;BASE ADDRESS OF ROM
100000      LINADR= ROMADR+ROMSIZ      ;BASE ADDRESS OF LINK MEMORY
000100      MINBC=  64.                ; 64 BYTES
002752      MAXBC=  1518.-4.           ;MAXIMUM # OF BYTES IN A TRANSMIT BUFFER
000000      DATERR= 0                  ;FLAG INDICATING DATA ERROR OCCURRED
000001      ADRERR= 1                  ;FLAG INDICATING ADDRESS ERROR OCCURRED
            ;
177777      INITH   =       -1         ; MINUS ONE     ( INITIAL CRC VALUE)
177777      INITL   =       -1
166670      POLYH   =       166670           ; FUNCTION POLYNOMIAL HIGH WORD
101440      POLYL   =       101440           ; FUNCTION POLYNOMIAL LOW WORD
157273      EXCRCH  =       157273           ; EXPECTED CRC HIGH
020343      EXCRCL  =       020343           ; EXPECTED CRC LOW
```

```
                                        .SBTTL  F_MODULE MICROCODE

000000' 106427  000340          MICROF::MTPS    #PRI07                  ;DISABLE INTERRUPTS
000004' 012737  000000  177776          MOV     #0,@#CMDREG             ;TURN OFF THE LINK
000012' 012706  001000                  MOV     #STACK,SP               ;SETUP STACK
000016' 112767  000001  000604          MOVB    #INMON,PCSR1            ;TELL HOST WE ARE IN MICROMONITOR
000024' 016737  000600  021020          MOV     PCSR1,@#IPCSR1
000032' 012737  004000  021000          MOV     #DNI,@#IPCSR0           ;TELL HOST THE LOAD AND START FINISHED
000040' 010700                          MOV     PC,R0                   ;GET ADDRESS OF UNEXPECTED ERROR...
000042' 062700  000404                  ADD     #ERRINT-.,R0            ;HANDLER
000046' 005001                          CLR     R1                      ;FILL ALL UNUSED VECTORS WITH TRAP...
000050' 010021                  10$:    MOV     R0,(R1)+                ;HANDLER
000052' 012721  000340                  MOV     #PRI07,(R1)+
000056' 020127  001000                  CMP     R1,#1000
000062' 002772                          BLT     10$

000064' 010700                          MOV     PC,R0                   ;SETUP PARITY TRAP VECTOR
000066' 062700  000462                  ADD     #PARINT-.,R0
000072' 010037  000140                  MOV     R0,@#PARVEC
000076' 012737  000340  000142          MOV     #PRI07,@#PARVEC+2

000104' 010700                          MOV     PC,R0                   ;SETUP DMA INTERRUPT VECTOR
000106' 062700  000364                  ADD     #DMAINT-.,R0
000112' 010037  000114                  MOV     R0,@#DMAVEC
000116' 012737  000340  000116          MOV     #PRI07,@#DMAVEC+2

000124' 010700                          MOV     PC,R0                   ;SETUP CSR WRITE VECTOR
000126' 062700  000310                  ADD     #CSRWRT-.,R0
000132' 010037  000064                  MOV     R0,@#CSRVEC
000136' 012737  000200  000066          MOV     #PRI04,@#CSRVEC+2

000144' 010700                          MOV     PC,R0                   ;SETUP SANTITY TIMER VECTOR
000146' 062700  000316                  ADD     #TIMINT-.,R0
000152' 010037  000134                  MOV     R0,@#SANVEC
000156' 012737  000240  000136          MOV     #PRI05,@#SANVEC+2

000164' 010700                          MOV     PC,R0                   ;SETUP TRANSMITTER VECTOR
000166' 062700  000414                  ADD     #TRNINT-.,R0
000172' 010037  000070                  MOV     R0,@#TRNVEC
000176' 012737  000200  000072          MOV     #PRI04,@#TRNVEC+2

000204' 010700                          MOV     PC,R0                   ;SETUP RECEIVER VECTOR
000206' 062700  000360                  ADD     #RCVINT-.,R0
000212' 010037  000120                  MOV     R0,@#RCVVEC
000216' 012737  000240  000122          MOV     #PRI05,@#RCVVEC+2

000224' 013700  021040                  MOV     @#PCSRSW,R0             ;GET SWITCH PACK BITS
000230' 052700  176000                  BIS     #176000,R0              ;MAP THEM INTO HOST I/O PAGE
000234' 006300                          ASL     R0                      ;SHIFT OVER TO POSITION CORRECTLY
000236' 006300                          ASL     R0
000240' 006300                          ASL     R0
000242' 062700  000004                  ADD     #4,R0                   ;PCSR2 IS PCSR0+4
000246' 010067  000360                  MOV     R0,IPCSR2               ;SAVE PCSR2 ADDRESS
000252' 012767  000003  000354          MOV     #3,IPCSR2+2             ;HIGH ORDER BITS 17:16
000260' 005067  000340                  CLR     FLG6                    ;INITIALIZE FLAG WORD
```

```
000264' 106427  000000           15$:    MTPS    #PRI00                  ;ALLOW INTERRUPTS

000270' 005767  000330           20$:    TST     FLG6                    ;WAIT FOR A COMMAND FROM HOST
000274' 001775                           BEQ     20$

000276' 106427  000340                   MTPS    #PRI07                  ;RAISE CPU PRIORITY TO SERVICE COMMAND
000302' 032767  000001  000314           BIT     #CSRFLG,FLG6            ;DID HOST GIVE US A COMMAND?
000310' 001001                           BNE     30$                     ;YES
000312' 000777                           BR      .                       ;NO, ERROR SO JUST SIT HERE...
                                                                         ;FOR LACK OF ANYTHING BETTER TO DO

000314' 113700  021000           30$:    MOVB    @#IPCSR0,R0             ;GET WHAT HOST WROTE TO PCSR0
000320' 042700  177760                   BIC     #177760,R0              ;STRIP ALL BUT COMMAND BITS
000324' 001004                           BNE     35$                     ;WAS IT THE CLEAR FUNCTION?
000326' 012737  000001  021020           MOV     #INMON,@#IPCSR1         ;YES, CLEAR OUT THE TEST # BITS
000334' 000432                           BR      50$
000336' 022700  000017           35$:    CMP     #17,R0                  ;RESTART OPERATIONAL MICROCODE?
000342' 001432                           BEQ     60$                     ;YES
000344' 162700  000001                   SUB     #1,R0
000350' 010701                           MOV     PC,R1                   ;GET ADDRESS OF OUR COMMAND TABLE
000352' 062701  000240                   ADD     #TBLD-.,R1
000356' 006300                           ASL     R0                      ;MAKE COMMAND A BYTE OFFSET
000360' 060001                           ADD     R0,R1                   ;USE IT TO INDEX INTO COMMAND TABLE
000362' 061101                           ADD     (R1),R1                 ;R1 NOW HAS COMMAND ROUTINE ADDRESS
000364' 004711                           JSR     PC,(R1)                 ;EXECUTE AS COMMANDED FROM HOST
000366' 103404                           BCS     40$                     ;ERROR OCCURRED
000370' 112767  000001  000232           MOVB    #INMON,PCSR1            ;INDICATE TO HOST WE ARE BACK IN...
000376' 000403                           BR      45$                     ;MICROMONITR
000400' 112767  000003  000222   40$:    MOVB    #INERR,PCSR1            ;INDICATE TO HOST ERROR OCCURRED
000406' 016737  000216  021020   45$:    MOV     PCSR1,@#IPCSR1
000414' 012737  004000  021000           MOV     #DNI,@#IPCSR0           ;TELL HOST THIS MICROTEST FINISHED
000422' 005067  000176           50$:    CLR     FLG6                    ;RESET FLAG WORD
000426' 000716                           BR      15$                     ;GO WAIT FOR ANOTHER COMMAND

000430' 005000                   60$:    CLR     R0                      ;FAKE SUCCESSFUL SELF TEST RESULTS
000432' 000137  040006                   JMP     @#40006                 ;START OPERATIONAL MICROCODE

000436' 052767  000001  000160   CSRWRT: BIS     #CSRFLG,FLG6            ;INDICATE A CSR WRITE INTERRUPT OCCURED
000444' 000002                           RTI

000446' 052767  000002  000150   ERRINT: BIS     #ERRFLG,FLG6            ;INDICATE A UNEXPECTED INTERRUPT OCCURED
000454' 012737  020000  021000           MOV     #UNIERR,@#IPCSR0        ;TELL HOST AN UNEXPECTED INTERRUPT
                                                                         ;HAPPENED
                                                                         ;JUST SIT HERE AND SPIN WHEELS
000462' 000777                           BR      .                       ;COUNT ON HOST TO TIMEOUT

000464' 005267  000136           TIMINT: INC     SANTIM                  ;COUNT TICKS AS THEY OCCUR
000470' 000002                           RTI

000472' 013767  021002  000142   DMAINT: MOV     @#DMACSR,DMDONE         ;GET DMA STATUS
000500' 032767  040000  000134           BIT     #BIT14,DMDONE           ;DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
000506' 001404                           BEQ     10$                     ;NO
000510' 012737  040000  021000           MOV     #NXMERR,@#IPCSR0        ;YES, TELL HOST A NON-EXISTANT MEMORY
                                                                         ;LOCATION WAS ADDRESSED
000516' 000407                           BR      20$
000520' 032767  100000  000114   10$:    BIT     #BIT15,DMDONE           ;DID A NPR TIMEOUT OCCUR?
```

```
000526' 001407                                BEQ     30$                      ;NO
000530' 012737  100000  021000                MOV     #NPRERR,@#IPCSR0         ;TELL HOST NPR TIMEOUT HAPPENED
000536' 012737  100000  021002  20$:          MOV     #BIT15,@#DMACSR          ;CLEAR THE INTERRUPT IN THE DMA ENGINE
000544' 000777                                BR      .                        ;SIT HERE AND SPIN WHEELS
000546' 000002                  30$:          RTI

000550' 052767  000004  000046  PARINT: BIS   #PARFLG,FLG6             ;SET PARITY ERROR OCCURRED
000556' 012737  010000  021000                MOV     #PARERR,@#IPCSR0         ;TELL HOST A LINK MEMORY PARITY ERROR
                                                                               ;OCCURRED
                                                                               ;JUST SIT HERE AND SPIN WHEELS
000564' 000777                                BR      .                        ;COUNT ON HOST TO TIMEOUT


000566' 005737  021044          RCVINT: TST   @#LRBUF                  ;READ BUFFER DONE REGISTER...
                                                                               ;WHICH CLEARS THE INTERRUPT
000572' 052767  000100  000024                BIS     #RCVFLG,FLG6             ;SET RECEIVER INTERRUPT OCCURRED
000600' 000002                                RTI

000602' 052767  000040  000014  TRNINT: BIS   #TRNFLG,FLG6             ;SET TRANSMITTER INTERRUPT OCCURRED
000610' 000002                                RTI

000612' 000170                  TBLD:   .WORD   MICF1-.                  ; CRC DATA PATTERN
000614' 000460                          .WORD   MICF2-.                  ; CRC ERROR TEST
000616' 001032                          .WORD   MICF3-.                  ; CRC PATTERN LENGTH
000620' 001544                          .WORD   MICF4-.                  ; BUFFER RECOVERY TEST
000622' 002222                          .WORD   MICF5-.                  ;HALF-DUPLEX TEST

000624' 000000                  FLG6:   .WORD   0                        ;FLAG WORD
000626' 000000                  SANTIM: .WORD   0                        ;COUNT FOR SANITY TIMER
000630' 000000                  PCSR1:  .WORD   0                        ;COPY OF WHAT GOES TO PCSR1
000632' 000000  000000          IPCSR2: .WORD   0,0                      ;ADDRESS IN HOST MEMORY FOR PCSR2
000636' 000000  000000          PCBADR: .WORD   0,0                      ;ADDRESS IN HOST MEMORY FOR PCB
000642' 000000                  DMDONE: .WORD   0
000644' 000000                  RBUFP:  .WORD   0                        ;POINTER TO RECIEVE BUFFER
000646' 000000                  TBUFP:  .WORD   0                        ;POINTER TO XMIT BUFFER
000650' 000000                  CRC1:   .WORD   0                        ;HOLDS LAST DATA BYTE AND 1ST CRC BYTE
000652' 000000                  CRC23:  .WORD   0                        ;HOLDS 2ND AND 3RD CRC BYTES
000654' 000000                  CRC4:   .WORD   0                        ;HOLDS 4TH CRC BYTE
```

```
                                 ;
                                 ;
                                 ;  ************************
                                 ;  ***** SUBROUTINES *****
                                 ;  ************************
                                 ;
000656' 106746                   BLKCRC: MFPS    -(SP)                   ; SAVE PSW
000660' 010046                           MOV     R0,-(SP)                ; SAVE R0
000662' 010146                           MOV     R1,-(SP)
000664' 010246                           MOV     R2,-(SP)
000666' 012704  177777                   MOV     #INITH,R4               ; INITIAL CRC HIGH WORD
000672' 012705  177777                   MOV     #INITL,R5               ; INITIAL CRC LOW WORD
000676' 112001                   10$:    MOVB    (R0)+,R1                ; GET NEXT BYTE
000700' 004767  000014                   JSR     PC,GETCRC               ; CALCULATE CRC
000704' 077204                   20$:    SOB     R2,10$                  ; LOOP TILL DONE
000706' 012602                           MOV     (SP)+,R2
000710' 012601                           MOV     (SP)+,R1
000712' 012600                           MOV     (SP)+,R0
000714' 106426                           MTPS    (SP)+
000716' 000207                           RTS     PC
                                 ;
                                 ;
000720' 106746                   GETCRC: MFPS    -(SP)                   ; SAVE REGISTERS
000722' 010146                           MOV     R1,-(SP)
000724' 010246                           MOV     R2,-(SP)
000726' 010346                           MOV     R3,-(SP)
000730' 042701  177400                   BIC     #^C377,R1               ; CLEAR HIGH BYTE
000734' 074105                           XOR     R1,R5                   ; MERGE NEW BYTE WITH OLD CRC
000736' 012702  166670                   MOV     #POLYH,R2               ; GET POLYNOMIAL HIGH WORD
000742' 012703  101440                   MOV     #POLYL,R3               ; GET CRC POLYNOMIAL LOW WORD
000746' 012701  000010                   MOV     #8.,R1                  ; LOOP COUNT
000752' 000241                   10$:    CLC                             ; CLEAR THE CARRY
000754' 006004                           ROR     R4                      ; SHIFT RIGHT THE CRC
000756' 006005                           ROR     R5                      ; 32 BITS WORTH
000760' 103002                           BCC     20$                     ; SKIP IF BIT 0 NOT SET
000762' 074204                           XOR     R2,R4                   ; EXCLUSIVE OR IN THE POLY
000764' 074305                           XOR     R3,R5                   ; BOTH HIGH AND LOW WORDS
000766' 077107                   20$:    SOB     R1,10$                  ; AND LOOP ON ALL 8 BITS
000770' 012603                           MOV     (SP)+,R3                ; RESTORE REGISTERS
000772' 012602                           MOV     (SP)+,R2
000774' 012601                           MOV     (SP)+,R1
000776' 106426                           MTPS    (SP)+
001000' 000207                           RTS     PC
                                 ;
                                 ;
```

```
                                       ;
                                       .SBTTL  CRC DATA PATTERN TEST
                                       ;
                                       ; ***********************************
                                       ; ***** TELL HIM WE ARE TESTING *****
                                       ; ***********************************
                                       ;
001002' 112767  000002  177620  MICF1: MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
001010' 016737  177614  021020         MOV     PCSR1,@#IPCSR1
                                       ;
                                       ; *******************************************
                                       ; ***** RETRIEVE PATTERN FROM HOST MEMORY *****
                                       ; *******************************************
                                       ;
001016' 016737  177610  021010         MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB ADDRESS
001024' 016737  177604  021012         MOV     IPCSR2+2,@#MDMA1
001032' 013700  021014                 MOV     @#MDMAR0,R0             ; R5 NOW CONTAINS PCBB LOW
001036' 013701  021014                 MOV     @#MDMAR0,R1             ; R1 NOW CONTAINS PCBB HIGH
001042' 010037  021010                 MOV     R0,@#MDMA0              ; POINT AT PCBB
001046' 010137  021012                 MOV     R1,@#MDMA1
001052' 013703  021014                 MOV     @#MDMAR0,R3             ; R3 NOW HOLDS DATA PATTERN
                                       ;
                                       ;
                                       ; *****************************************************
                                       ; ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
                                       ; *****************************************************
                                       ;
001056' 012700  100000                 MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
001062' 010067  177556                 MOV     R0,RBUFP
001066' 005020                  10$:   CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
001070' 020027  104000                 CMP     R0,#LINADR+SIZ1K        ; STOP AT THE TOP
001074' 103774                         BLO     10$
                                       ;
                                       ;
                                       ; ********************************************
                                       ; ***** FILL XMIT BUFFER WITH TEST PATTERN *****
                                       ; ********************************************
                                       ;
001076' 010067  177544                 MOV     R0,TBUFP                ; SAVE COPY OF ADDRESS
001102' 010320                  20$:   MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
001104' 020027  110000                 CMP     R0,#LINADR+SIZ2K
001110' 103774                         BLO     20$
                                       ;
                                       ;
                                       ; ********************************************
                                       ; ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
                                       ; ********************************************
                                       ;
001112' 012737  000200  177776         MOV     #MODE,@#CMDREG          ; ENABLE LINK, SELECT MODE REG
001120' 012737  100004  177774         MOV     #PROM!LOOP,@#MODREG     ; PROM MODE AND LOOPBACK
001126' 012737  100000  177776         MOV     #ENABLE,@#CMDREG        ; ENABLE IT
                                       ;
001134' 016701  177506                 MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
001140' 005021                         CLR     (R1)+                   ; CLEAR OUT STATUS WORD
001142' 012721  002752                 MOV     #MAXBC,(R1)+            ; CLEAR OUT THE FIFO
001146' 005037  021034                 CLR     @#CLRFIF                ; CLEAR THE FIFO
```

```
001152' 005067  177446                  CLR     FLG6
001156' 016737  177462  021032          MOV     RBUFP,@#LFRBUF          ; TELL DEUNA RECEIVE BUF LOC
001164' 016737  177456  021030          MOV     TBUFP,@#LTAC            ; TELL DEUNA WHERE XMIT BUF IS
                                ;
001172' 106427  000140                  MTPS    #PRI03                  ; ALLOW INTERRUPTS
001176' 032767  000100  177420  30$:    BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
001204' 001774                          BEQ     30$
                                ;
001206' 032767  000040  177410  40$:    BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INTERRUPT TOO
001214' 001774                          BEQ     40$
                                ;
001216' 106427  000340                  MTPS    #PRI07                  ; DISABLE INTERRUPTS
                                ;
                                ;
                                ; ***********************************************************
                                ; ***** CALCULATE THE CRC ON THE RECEIVER BUFFER CONTENTS *****
                                ; ***********************************************************
                                ;
001222' 016700  177416                  MOV     RBUFP,R0                ; POINT AT RECEIVE BUFFER
001226' 062700  000004                  ADD     #4,R0                   ; BY-PASS THE STATUS AND BC
001232' 012702  002756                  MOV     #MAXBC+4,R2             ; INCLUDE CRC BYTES
001236' 004767  177414                  JSR     PC,BLKCRC
001242' 020427  157273                  CMP     R4,#EXCRCH
001246' 001010                          BNE     50$                     ; ERROR IF NOT
001250' 020527  020343                  CMP     R5,#EXCRCL
001254' 001005                          BNE     50$
001256' 000241                          CLC                             ;INDICATE SUCCESS
001260' 112767  000001  177343  45$:    MOVB    #1,PCSR1+1              ; TELL WHAT TEST WE JUST FINISHED
001266' 000207                          RTS     PC
                                ;
                                ;
                                ; ***********************
                                ; ***** ERROR EXIT *****
                                ; ***********************
                                ;
001270' 000261                  50$:    SEC                             ; TELL HIM WE MADE ERROR
001272' 000772                          BR      45$                     ; LEAVE
```

```
                                         ;
                                         .SBTTL  CRC ERROR TEST
                                         ;
                                         ;  ***************************************
                                         ;  ***** TELL HIM WE ARE TESTING *****
                                         ;  ***************************************
                                         ;
001274' 112767  000002  177326  MICF2:   MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
001302' 016737  177322  021020           MOV     PCSR1,@#IPCSR1
                                         ;
                                         ;
                                         ;  *********************************************
                                         ;  ***** RETRIEVE PATTERN FROM HOST MEMORY *****
                                         ;  *********************************************
                                         ;
001310' 016737  177316  021010           MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB ADDRESS
001316' 016737  177312  021012           MOV     IPCSR2+2,@#MDMA1
001324' 013704  021014                   MOV     @#MDMAR0,R4             ; R4 NOW CONTAINS PCBB LOW
001330' 013705  021014                   MOV     @#MDMAR0,R5             ; R5 NOW CONTAINS PCBB HIGH
001334' 010437  021010                   MOV     R4,@#MDMA0              ; POINT AT PCBB
001340' 010537  021012                   MOV     R5,@#MDMA1
001344' 013703  021014                   MOV     @#MDMAR0,R3             ; R3 NOW HOLDS DATA PATTERN
                                         ;
                                         ;
                                         ;  ****************************************************
                                         ;  ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
                                         ;  ****************************************************
                                         ;
001350' 012700  100000                   MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
001354' 010067  177264                   MOV     R0,RBUFP
001360' 005020                   10$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
001362' 020027  104000                   CMP     R0,#LINADR+SIZ1K        ; STOP AT THE TOP
001366' 103774                           BLO     10$
                                         ;
                                         ;
                                         ;  **********************************************
                                         ;  ***** FILL XMIT BUFFER WITH TEST PATTERN *****
                                         ;  **********************************************
                                         ;
001370' 010067  177252                   MOV     R0,TBUFP                ; SAVE COPY OF ADDRESS
001374' 010320                   20$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
001376' 020027  110000                   CMP     R0,#LINADR+SIZ2K
001402' 103774                           BLO     20$
                                         ;
                                         ;
                                         ;  *******************************************
                                         ;  ***** CALCULATE CRC ON TRANSMIT BUFFER *****
                                         ;  *******************************************
                                         ;
001404' 016700  177236                   MOV     TBUFP,R0                ; POINT AT RECEIVE BUFFER
001410' 062700  000004                   ADD     #4,R0                   ; SKIP 1ST WORD AND BYTE COUNT
001414' 012702  002746                   MOV     #MAXBC-4,R2             ; DO A BUNCH
001420' 004767  177232                   JSR     PC,BLKCRC
001424' 016700  177216                   MOV     TBUFP,R0                ; POINT AT XMIT BUFFER
001430' 062700  000004                   ADD     #4,R0                   ; BY-PASS STATUS AND BC
001434' 062700  002746                   ADD     #MAXBC-4,R0             ; ADD BUFFER OFFSET
```

```
001440' 005104                          COM     R4                      ; MUST SEND THE COMPLEMENT
001442' 005105                          COM     R5
001444' 005205                          INC     R5                      ; INTRODUCE AN ERROR IN CRC!
001446' 010520                          MOV     R5,(R0)+                ; APPEND THE CRC WORD
001450' 010420                          MOV     R4,(R0)+                ; APPEND THE OTHER CRC WORD
                                ;
                                ;
                                ;       *********************************************
                                ;       ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
                                ;       *********************************************
                                ;
001452' 012737  000200  177776          MOV     #MODE,@#CMDREG                  ; ENABLE, SELECT MODE
001460' 012737  100014  177774          MOV     #PROM!LOOP!DTCR,@#MODREG        ; PROM MODE AND LOOPBACK
001466' 012737  100000  177776          MOV     #ENABLE,@#CMDREG                ; ENABLE THE XMTR
                                ;
001474' 016701  177146                  MOV     TBUFP,R1                ; POINT AT XMIT BUFFER
001500' 005021                          CLR     (R1)+                   ; CLEAR OUT STATUS WORD
001502' 012721  002752                  MOV     #MAXBC,(R1)+            ; CLEAR OUT THE FIFO
001506' 005037  021034                  CLR     @#CLRFIF                ; CLEAR THE FIFO
001512' 005067  177106                  CLR     FLG6
001516' 016737  177122  021032          MOV     RBUFP,@#LFRBUF          ; TELL DEUNA RECEIVE BUF LOC
001524' 016737  177116  021030          MOV     TBUFP,@#LTAC            ; TELL DEUNA WHERE XMIT BUF IS
                                ;
001532' 106427  000140                  MTPS    #PRI03                  ; ALLOW INTERRUPTS
001536' 032767  000100  177060  30$:    BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
001544' 001774                          BEQ     30$
                                ;
001546' 032767  000040  177050  40$:    BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INTERRUPT TOO
001554' 001774                          BEQ     40$
                                ;
001556' 106427  000340                  MTPS    #PRI07                  ; DISABLE INTERRUPTS
                                ;
                                ;
                                ;       *********************************************
                                ;       ***** WRITE STATUS REGISTER TO HOST MEMORY *****
                                ;       *********************************************
                                ;
001562' 012700  100000                  MOV     #LINADR,R0              ; GET RECEIVE STATUS WORD
001566' 011003                          MOV     (R0),R3                 ; GET STATUS WORD
                                ;
001570' 016737  177036  021010          MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB
001576' 016737  177032  021012          MOV     IPCSR2+2,@#MDMA1
001604' 013700  021014                  MOV     @#MDMAR0,R0             ; R0 NOW CONTAINS PCBB LOW
001610' 013701  021014                  MOV     @#MDMAR0,R1             ; R1 NOW CONTAINS PCBB HI
001614' 062700  000002                  ADD     #2,R0                   ; BUMP TO NEXT HOST WORD
001620' 005501                          ADC     R1
001622' 010037  021010                  MOV     R0,@#MDMA0              ; POINT TO PCBB
001626' 010137  021012                  MOV     R1,@#MDMA1
001632' 010337  021026                  MOV     R3,@#MDMAW0             ; WRITE STATUS WORD TO HOST
                                ;
                                ;
001636' 112767  000002  176765          MOVB    #2,PCSR1+1              ; TELL HIM WHICH TEST IT IS
001644' 000241                          CLC
001646' 000207                          RTS     PC
```

```
                                         ;
                                         .SBTTL  CRC PATTERN LENGTH TEST
                                         ;
                                         ;  ************************************
                                         ;  ***** TELL HIM WE ARE TESTING *****
                                         ;  ************************************
                                         ;
001650' 112767  000002  176752  MICF3:   MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
001656' 016737  176746  021020           MOV     PCSR1,@#IPCSR1
                                         ;
                                         ;
                                         ;  ********************************************
                                         ;  ***** RETRIEVE PATTERN FROM HOST MEMORY *****
                                         ;  ********************************************
                                         ;
001664' 016737  176742  021010           MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB ADDRESS
001672' 016737  176736  021012           MOV     IPCSR2+2,@#MDMA1
001700' 013704  021014                   MOV     @#MDMAR0,R4             ; R4 NOW CONTAINS PCBB LOW
001704' 013705  021014                   MOV     @#MDMAR0,R5             ; R5 NOW CONTAINS PCBB HIGH
001710' 010437  021010                   MOV     R4,@#MDMA0              ; POINT AT PCBB
001714' 010537  021012                   MOV     R5,@#MDMA1
001720' 013703  021014                   MOV     @#MDMAR0,R3             ; R3 NOW HOLDS DATA PATTERN
                                         ;
                                         ;
                                         ;  ****************************************************
                                         ;  ***** FILL RECEIVE BUFFER WITH BACKGROUND PATTERN *****
                                         ;  ****************************************************
                                         ;
001724' 012700  100000                   MOV     #LINADR,R0              ; RECEIVE BUFFER STARTS HERE
001730' 010067  176710                   MOV     R0,RBUFP
001734' 005020                   10$:    CLR     (R0)+                   ; FILL RECEIVE BUFFER WITH ZEROS
001736' 020027  104000                   CMP     R0,#LINADR+SIZ1K        ; STOP AT THE TOP
001742' 103774                           BLO     10$
                                         ;
                                         ;
                                         ;  ********************************************
                                         ;  ***** FILL XMIT BUFFER WITH TEST PATTERN *****
                                         ;  ********************************************
                                         ;
001744' 010067  176676                   MOV     R0,TBUFP                ; SAVE COPY OF ADDRESS
001750' 010320                   20$:    MOV     R3,(R0)+                ; FILL XMIT BUFFER WITH PATTERN
001752' 020027  110000                   CMP     R0,#LINADR+SIZ2K
001756' 103774                           BLO     20$
                                         ;
                                         ;
                                         ;  **********************************************
                                         ;  ***** RETRIEVE BYTE COUNT FROM HOST MEMORY *****
                                         ;  **********************************************
                                         ;
001760' 016737  176646  021010           MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB
001766' 016737  176642  021012           MOV     IPCSR2+2,@#MDMA1
001774' 013704  021014                   MOV     @#MDMAR0,R4             ; R4 NOW CONTAINS PCBB LO
002000' 013705  021014                   MOV     @#MDMAR0,R5             ; R5 NOW CONTAINS PCBB HI
002004' 062704  000002                   ADD     #2,R4                   ; BUMP LOW BY TWO
002010' 005505                           ADC     R5
002012' 010437  021010                   MOV     R4,@#MDMA0              ; POINT AT PCBB
```

```
002016' 010537  021012                  MOV     R5,@#MDMA1
002022' 013703  021014                  MOV     @#MDMARO,R3                     ; R3 NOW HOLDS BYTE COUNT
                                ;
                                ;
                                ; ***************************************************
                                ; ***** CALCULATE CRC ON TRANSMIT BUFFER *****
                                ; ***************************************************
                                ;
002026' 016700  176614                  MOV     TBUFP,R0                        ; POINT AT RECEIVE BUFFER
002032' 062700  000004                  ADD     #4,R0                           ; BY-PASS STATUS AND BC
002036' 010302                          MOV     R3,R2                           ; R3 CONTAINS BYTE COUNT
002040' 004767  176612                  JSR     PC,BLKCRC
002044' 016700  176576                  MOV     TBUFP,R0                        ; POINT AT XMIT BUFFER
002050' 062700  000004                  ADD     #4,R0                           ; BY-PASS STATUS AND BC
002054' 060300                          ADD     R3,R0                           ; ADD BUFFER BYTE COUNT
002056' 005104                          COM     R4                              ; COMPLEMENT BEFORE SENDING
002060' 005105                          COM     R5
002062' 032700  000001                  BIT     #1,R0                           ;ARE WE LOOKING AT AN ODD BOUNDARY?
002066' 001427                          BEQ     25$                             ;NO, GOOD IT MAKES THINGS EASIER
                                ;
002070' 162700  000001                  SUB     #1,R0                           ;POINT BACK TO EVEN BOUNDARY
002074' 011067  176550                  MOV     (R0),CRC1                       ;GET DATA BYTE AND JUNK
002100' 110567  176545                  MOVB    R5,CRC1+1                       ;REPLACE JUNK WITH 1ST CRC BYTE
002104' 000305                          SWAB    R5                              ;POSITION 2ND CRC BYTE
002106' 110567  176540                  MOVB    R5,CRC23                        ;GET 2ND CRC BYTE
002112' 110467  176535                  MOVB    R4,CRC23+1                      ;GET 3RD CRC BYTE
002116' 000304                          SWAB    R4                              ;POSITION 4TH CRC BYTE
002120' 110467  176530                  MOVB    R4,CRC4                         ;GET 4TH CRC BYTE
002124' 105067  176525                  CLRB    CRC4+1                          ;CLEAR JUNK FOR THE HELL OF IT
002130' 016720  176514                  MOV     CRC1,(R0)+                      ;STORE DATA AND 1ST CRC BYTE IN BUFFER
002134' 016720  176512                  MOV     CRC23,(R0)+                     ;STORE 2ND AND 3RD CRC BYTES IN BUFFER
002140' 016710  176510                  MOV     CRC4,(R0)                       ;STORE 4TH CRC BYTE IN BUFFER
002144' 000402                          BR      26$                             ;CARRY ON

002146' 010520                  25$:    MOV     R5,(R0)+                        ; APPEND THE CRC WORD
002150' 010420                          MOV     R4,(R0)+                        ; APPEND THE OTHER CRC WORD
002152'                         26$:
                                ;
                                ;
                                ; ***************************************************
                                ; ***** SET UP LINK FOR DATAGRAM LOOPBACK *****
                                ; ***************************************************
                                ;
002152' 012737  000200  177776          MOV     #MODE,@#CMDREG                  ; ENABLE, SELECT MODE
002160' 012737  100014  177774          MOV     #PROM!LOOP!DTCR,@#MODREG        ; PROM MODE AND LOOPBACK
002166' 012737  100000  177776          MOV     #ENABLE,@#CMDREG                ; LET HER GO!
                                ;
002174' 016701  176446                  MOV     TBUFP,R1                        ; POINT AT XMIT BUFFER
002200' 005021                          CLR     (R1)+                           ; CLEAR OUT STATUS WORD
002202' 062703  000004                  ADD     #4,R3                           ; ACCOUNT FOR CRC BYTES
002206' 010321                          MOV     R3,(R1)+                        ; WRITE THE BYTE COUNT
002210' 005037  021034                  CLR     @#CLRFIF                        ; CLEAR THE FIFO
002214' 005067  176404                  CLR     FLG6
002220' 016737  176420  021032          MOV     RBUFP,@#LFRBUF                  ; TELL DEUNA RECEIVE BUF LOC
002226' 016737  176414  021030          MOV     TBUFP,@#LTAC                    ; TELL DEUNA WHERE XMIT BUF IS
                                ;
```

```
002234' 106427  000140                  MTPS    #PRI03                  ; ALLOW INTERRUPTS
002240' 032767  000100  176356  30$:    BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
002246' 001774                          BEQ     30$
                                ;
002250' 032767  000040  176346  40$:    BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INTERRUPT TOO
002256' 001774                          BEQ     40$
                                ;
002260' 106427  000340                  MTPS    #PRI07                  ; DISABLE INTERRUPTS
                                ;
                                ;
                                ; ***********************************************
                                ; ***** WRITE STATUS REGISTER TO HOST MEMORY *****
                                ; ***********************************************
                                ;
002264' 016700  176354                  MOV     RBUFP,R0                ; GET RECEIVE STATUS WORD
002270' 011003                          MOV     (R0),R3                 ; GET STATUS WORD
                                ;
002272' 016737  176334  021010          MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB
002300' 016737  176330  021012          MOV     IPCSR2+2,@#MDMA1
002306' 013700  021014                  MOV     @#MDMAR0,R0             ; R0 NOW CONTAINS PCBB LOW
002312' 013701  021014                  MOV     @#MDMAR0,R1             ; R1 NOW CONTAINS PCBB HI
002316' 062700  000004                  ADD     #4,R0                   ; BUMP TO NEXT HOST WORD
002322' 005501                          ADC     R1
002324' 010037  021010                  MOV     R0,@#MDMA0              ; POINT TO PCBB
002330' 010137  021012                  MOV     R1,@#MDMA1
002334' 010337  021026                  MOV     R3,@#MDMAW0             ; WRITE STATUS WORD TO HOST
                                ;
                                ;
002340' 016702  176302                  MOV     TBUFP,R2                ; GET XMIT BUFFER STATUS
002344' 011203                          MOV     (R2),R3
002346' 010337  021026                  MOV     R3,@#MDMAW0             ; WRITE XMIT STATUS TO HOST
002352' 112767  000003  176251          MOVB    #3,PCSR1+1              ; TELL HIM WHICH TEST IT IS
002360' 000241                          CLC
002362' 000207                          RTS     PC
```

```
                                        ;
                                        .SBTTL  RECEIVE BUFFER RECOVERY - RUNT TEST
                                        ;
                                        ;  **************************************
                                        ;  ***** TELL HIM WE ARE TESTING *****
                                        ;  **************************************
                                        ;
002364' 112767  000002  176236  MICF4:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
002372' 016737  176232  021020          MOV     PCSR1,@#IPCSR1
                                        ;
002400' 005067  176222                  CLR     SANTIM                  ; CLEAR FLAG FOR TIMER
                                        ;
                                        ;
                                        ;  **************************************
                                        ;  ***** CLEAR THE RECEIVE BUFFER *****
                                        ;  **************************************
                                        ;
002404' 012700  100000                  MOV     #LINADR,R0              ; RECEIVER BUFFER STARTS HERE
002410' 010067  176230                  MOV     R0,RBUFP
002414' 005020                  10$:    CLR     (R0)+                   ; FILL BUFFER WITH ZEROS
002416' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ; STOP AT THE TOP
002422' 103774                          BLO     10$
                                        ;
                                        ;
                                        ;  **************************************
                                        ;  ***** FILL XMIT WITH RUNT MARKER *****
                                        ;  **************************************
                                        ;
002424' 010067  176216                  MOV     R0,TBUFP                ; SAVE COPY OF THE ADDRESS
002430' 012720  125252          20$:    MOV     #125252,(R0)+
002434' 020027  110000                  CMP     R0,#LINADR+SIZ2K
002440' 103773                          BLO     20$
                                        ;
                                        ;
                                        ;  ************************************************
                                        ;  ***** RETRIEVE BYTE COUNT FROM HOST MEMORY *****
                                        ;  ************************************************
                                        ;
002442' 016737  176164  021010          MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB ADDRESS
002450' 016737  176160  021012          MOV     IPCSR2+2,@#MDMA1
002456' 013700  021014                  MOV     @#MDMAR0,R0             ; R4 NOW CONTAINS PCBB LOW
002462' 013701  021014                  MOV     @#MDMAR0,R1             ; R5 NOW CONTAINS PCBB HIGH
002466' 010037  021010                  MOV     R0,@#MDMA0              ; POINT AT PCBB
002472' 010137  021012                  MOV     R1,@#MDMA1
002476' 013703  021014                  MOV     @#MDMAR0,R3             ; R3 NOW HOLDS BYTE COUNT
                                        ;
                                        ;
                                        ;  ************************************************
                                        ;  ***** SET UP LINK FOR RUNT DATAGRAM LOOPBACK *****
                                        ;  ************************************************
                                        ;
002502' 012737  000200  177776          MOV     #MODE,@#CMDREG  ; DISABLE XMIT CRC - TRUE BC
002510' 012737  100014  177774          MOV     #PROM!LOOP!DTCR,@#MODREG
002516' 012737  100000  177776          MOV     #ENABLE,@#CMDREG
                                        ;
002524' 016704  176116                  MOV     TBUFP,R4                ; POINT AT XMIT BUFFER
```

```
002530' 005024                          CLR     (R4)+                   ; CLEAR OUT STATUS WORD
002532' 010324                          MOV     R3,(R4)+                ; WRITE PASSED BYTE COUNT
                                ;
002534' 005037  021034                  CLR     @#CLRFIF                ; CLEAR THE FIFO
002540' 005067  176060                  CLR     FLG6                    ; CLEAR THE INTERRUPT FLAG
002544' 016737  176074  021032          MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
002552' 016737  176070  021030          MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                ;
002560' 106427  000140                  MTPS    #PRI03                  ; ALLOW XMITTER TO INTERRUPT
002564' 026727  176036  000002  30$:    CMP     SANTIM,#2               ; WAIT AT MOST 2 SECONDS
002572' 002012                          BGE     50$                     ; EXIT NORMALLY IF TIMER DONE
002574' 032767  000100  176022          BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
002602' 001770                          BEQ     30$                     ; IF NONE, WERE OK
                                ;
                                ;
                                ; ****************************
                                ; ***** ERROR FALLTHROUGH *****
                                ; ****************************
                                ;
002604' 032767  000040  176012  40$:    BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INT TOO!
002612' 001774                          BEQ     40$
                                ;
002614' 000261                          SEC                             ; TELL HOST UNA SCREWED UP!
002616' 000506                          BR      90$                     ; EXIT WITHOUT WRITING PTR
                                ;
                                ;
                                ; ****************************************
                                ; ***** NOW TRY LEGITIMATE BUFFER SIZE *****
                                ; ****************************************
                                ;
002620' 016700  176022          50$:    MOV     TBUFP,R0                ; GET TRANSMIT BUFFER POINTER
002624' 012720  052525          60$:    MOV     #52525,(R0)+
002630' 020027  110000                  CMP     R0,#LINADR+SIZ2K
002634' 103773                          BLO     60$
                                ;
                                ;
                                ; **************************************************
                                ; ***** SET UP LINK FOR MINIMUM DATAGRAM MESSAGE *****
                                ; **************************************************
                                ;
002636' 012737  000200  177776          MOV     #MODE,@#CMDREG          ; ENABLE LINK, SEL MODE REG
002644' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG
002652' 012737  100000  177776          MOV     #ENABLE,@#CMDREG
                                ;
002660' 016704  175762                  MOV     TBUFP,R4                ; POINT AT XMIT BUFFER
002664' 005024                          CLR     (R4)+                   ; CLEAR STATUS WORD
002666' 012724  000100                  MOV     #MINBC,(R4)+            ; SEND SMALL DATAGRAM
                                ;
002672' 005067  175726                  CLR     FLG6
002676' 016737  175742  021032          MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
002704' 016737  175736  021030          MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                ;
002712' 106427  000140                  MTPS    #PRI03                  ; ALLOW XMITR TO INTERRUPT
002716' 032767  000100  175700  70$:    BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
002724' 001774                          BEQ     70$
                                ;
```

```
002726' 032767  000040  175670  80$:    BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INT TOO!
002734' 001774                          BEQ     80$
                                ;
002736' 106427  000340                  MTPS    #PRI07                  ; DISABLE INTERRUPTS
                                ;
                                ;
                                ; ******************************************
                                ; ***** WRITE PARAMETERS TO HOST PCBB *****
                                ; ******************************************
                                ;
002742' 013703  021044                  MOV     @#LRBUF,R3              ; GET LINK POINTER
                                ;
002746' 016700  175672                  MOV     RBUFP,R0                ; POINT AT RECEIVER BUFFER
002752' 062700  000004                  ADD     #4,R0
002756' 011004                          MOV     (R0),R4                 ; GET PATTERN WORD
                                ;
                                ;
002760' 016737  175646  021010          MOV     IPCSR2,@#MDMA0          ; SET TO GET HOST PCBB
002766' 016737  175642  021012          MOV     IPCSR2+2,@#MDMA1
002774' 013700  021014                  MOV     @#MDMAR0,R0             ; R0 NOW CONTAINS PCBB LO
003000' 013701  021014                  MOV     @#MDMAR0,R1             ; R1 NOW CONTAINS PCBB HI
003004' 062700  000002                  ADD     #2,R0
003010' 005501                          ADC     R1
003012' 010037  021010                  MOV     R0,@#MDMA0              ; POINT AT PCBB+2
003016' 010137  021012                  MOV     R1,@#MDMA1
                                ;
003022' 010337  021026                  MOV     R3,@#MDMAW0             ; WRITE LINK POINTER
003026' 010437  021026                  MOV     R4,@#MDMAW0             ; WRITE DATA BYTE
                                ;
                                ;
003032' 000241                          CLC
003034' 112767  000004  175567  90$:    MOVB    #4,PCSR1+1              ; TELL HOST WE ARE DONE
003042' 000207                          RTS     PC
```

```
                                        .SBTTL  HALF-DUPLEX TEST
                                        ;
                                        ;  ************************************
                                        ;  ***** TELL HIM WE ARE TESTING *****
                                        ;  ************************************
                                        ;
003044' 112767  000002  175556  MICF5:  MOVB    #INTST,PCSR1            ; TELL HOST WE ARE TESTING
003052' 016737  175552  021020          MOV     PCSR1,@#IPCSR1
                                        ;
003060' 005067  175542                  CLR     SANTIM                  ; CLEAR FLAG FOR TIMER
                                        ;
                                        ;  ************************************
                                        ;  ***** CLEAR THE RECEIVE BUFFER *****
                                        ;  ************************************
                                        ;
003064' 012700  100000                  MOV     #LINADR,R0              ; RECEIVER BUFFER STARTS HERE
003070' 010067  175550                  MOV     R0,RBUFP
003074' 005020                  10$:    CLR     (R0)+                   ; FILL BUFFER WITH ZEROS
003076' 020027  104000                  CMP     R0,#LINADR+SIZ1K        ; STOP AT THE TOP
003102' 103774                          BLO     10$
                                        ;
                                        ;  *************************************
                                        ;  ***** FILL XMIT WITH RUNT MARKER *****
                                        ;  *************************************
                                        ;
003104' 010067  175536                  MOV     R0,TBUFP                ; SAVE COPY OF THE ADDRESS
003110' 012720  125252          20$:    MOV     #125252,(R0)+
003114' 020027  110000                  CMP     R0,#LINADR+SIZ2K
003120' 103773                          BLO     20$
                                        ;
                                        ;  ***************************************************
                                        ;  ***** SET UP LINK FOR RUNT DATAGRAM LOOPBACK *****
                                        ;  ***************************************************
                                        ;
003122' 012737  000200  177776          MOV     #MODE,@#CMDREG          ; DISABLE XMIT CRC - TRUE BC
003130' 012737  100005  177774          MOV     #PROM!LOOP!HDX,@#MODREG
003136' 012737  100000  177776          MOV     #ENABLE,@#CMDREG
                                        ;
003144' 016704  175476                  MOV     TBUFP,R4                ; POINT AT XMIT BUFFER
003150' 005024                          CLR     (R4)+                   ; CLEAR OUT STATUS WORD
003152' 010324                          MOV     R3,(R4)+                ; WRITE PASSED BYTE COUNT
                                        ;
003154' 005037  021034                  CLR     @#CLRFIF                ; CLEAR THE FIFO
003160' 005067  175440                  CLR     FLG6                    ; CLEAR THE INTERRUPT FLAG
003164' 016737  175454  021032          MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
003172' 016737  175450  021030          MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                        ;
003200' 106427  000140                  MTPS    #PRI03                  ; ALLOW XMITTER TO INTERRUPT
003204' 026727  175416  000002  30$:    CMP     SANTIM,#2               ; WAIT AT MOST 2 SECONDS
003212' 002012                          BGE     50$                     ; EXIT NORMALLY IF TIMER DONE
003214' 032767  000100  175402          BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
003222' 001770                          BEQ     30$                     ; IF NONE, WERE OK
                                        ;
```

```
                                        ;
                                        ;       *******************************
                                        ;       ***** ERROR FALLTHROUGH *****
                                        ;       *******************************
                                        ;
003224' 032767  000040  175372  40$:    BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INT TOO!
003232' 001774                          BEQ     40$
                                        ;
003234' 000261                          SEC                             ; TELL HOST UNA SCREWED UP!
003236' 000503                          BR      90$                     ; EXIT WITHOUT WRITING PTR
                                        ;
                                        ;
                                        ;       ************************************
                                        ;       ***** NOW SEE IF BUFFER RECOVERS *****
                                        ;       ************************************
                                        ;
003240' 016700  175402          50$:    MOV     TBUFP,R0                ; GET TRANSMIT BUFFER POINTER
003244' 012720  052525          60$:    MOV     #52525,(R0)+
003250' 020027  110000                  CMP     R0,#LINADR+SIZ2K
003254' 103773                          BLO     60$
                                        ;
                                        ;
                                        ;       ************************************************
                                        ;       ***** SET UP LINK FOR MINIMUM DATAGRAM MESSAGE *****
                                        ;       ************************************************
                                        ;
003256' 012737  000200  177776          MOV     #MODE,@#CMDREG          ; ENABLE LINK, SEL MODE REG
003264' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG
003272' 012737  100000  177776          MOV     #ENABLE,@#CMDREG
                                        ;
003300' 016704  175342                  MOV     TBUFP,R4                ; POINT AT XMIT BUFFER
003304' 005024                          CLR     (R4)+                   ; CLEAR STATUS WORD
003306' 012724  000100                  MOV     #MINBC,(R4)+            ; SEND SMALL DATAGRAM
                                        ;
003312' 005067  175306                  CLR     FLG6
003316' 016737  175322  021032          MOV     RBUFP,@#LFRBUF          ; TELL UNA WHERE RECEIVE BUF IS
003324' 016737  175316  021030          MOV     TBUFP,@#LTAC            ; TELL UNA WHERE XMIT BUF IS
                                        ;
003332' 106427  000140                  MTPS    #PRI03                  ; ALLOW XMITR TO INTERRUPT
003336' 032767  000100  175260  70$:    BIT     #RCVFLG,FLG6            ; WAIT FOR RECEIVER INTERRUPT
003344' 001774                          BEQ     70$
                                        ;
003346' 032767  000040  175250  80$:    BIT     #TRNFLG,FLG6            ; WAIT FOR XMIT INT TOO!
003354' 001774                          BEQ     80$
                                        ;
003356' 106427  000340                  MTPS    #PRI07                  ; DISABLE INTERRUPTS
                                        ;
                                        ;
                                        ;       ************************************
                                        ;       ***** WRITE PARAMETERS TO HOST PCBB *****
                                        ;       ************************************
                                        ;
003362' 013703  021044                  MOV     @#LRBUF,R3              ; GET LINK POINTER
                                        ;
003366' 016700  175252                  MOV     RBUFP,R0                ; POINT AT RECEIVER BUFFER
003372' 062700  000004                  ADD     #4,R0
```

```
     003376' 011004                          MOV     (R0),R4                ; GET PATTERN WORD
                                     ;
                                     ;
     003400' 016737  175226  021010          MOV     IPCSR2,@#MDMA0         ; SET TO GET HOST PCBB
     003406' 016737  175222  021012          MOV     IPCSR2+2,@#MDMA1
     003414' 013700  021014                  MOV     @#MDMAR0,R0            ; R0 NOW CONTAINS PCBB LO
     003420' 013701  021014                  MOV     @#MDMAR0,R1            ; R1 NOW CONTAINS PCBB HI
     003424' 010037  021010                  MOV     R0,@#MDMA0             ; POINT AT PCBB+2
     003430' 010137  021012                  MOV     R1,@#MDMA1
                                     ;
     003434' 010337  021026                  MOV     R3,@#MDMAW0            ; WRITE LINK POINTER
     003440' 010437  021026                  MOV     R4,@#MDMAW0            ; WRITE DATA BYTE
                                     ;
                                     ;
     003444' 000241                          CLC
     003446' 112767  000005  175155  90$:    MOVB    #5,PCSR1+1             ; TELL HOST WE ARE DONE
     003454' 000207                          RTS     PC

     003456' 003460                  MICFSZ::MICFSZ-MICROF+2
                                     ;
             000001                  .END
```

```
ACLO   = 002000           87#
ADRERR= 000001           130#
ADRREG= 177774            34#
ARAM   = 000100           80#
BIT0   = 000001           54#     92     104
BIT1   = 000002           53#    105
BIT10  = 002000           44#     56      73      87
BIT11  = 004000           43#     72      86
BIT12  = 010000           42#     56      71     114
BIT13  = 020000           41#     70     113
BIT14  = 040000           40#     69      85     112     244
BIT15  = 100000           39#     68      78      84     111     249     252
BIT2   = 000004           52#     91     106
BIT3   = 000010           51#     90     107
BIT4   = 000020           50#     89     108
BIT5   = 000040           49#     88     109
BIT6   = 000100           48#     80     110
BIT7   = 000200           47#     79
BIT8   = 000400           46#     56      74
BIT9   = 001000           45#
BLKCRC   000656R   002   295#    411     480     605
CLRFIF= 021034            22#    389*    502*    645*    743*    874*
CMDREG= 177776            35#    142*    382*    384*    495*    497*    637*    639*    735*    737*    780*    782*    866*
                         868*    911*    913*
COLL   = 000020           89#
CRC1     000650R   002   284#    615*    616*    623
CRC23    000652R   002   285#    618*    619*    624
CRC4     000654R   002   286#    621*    622*    625
CSRFLG= 000001           104#    200     231
CSRVEC= 000064            97#    167*    168*
CSRWRT   000436R   002   166     231#
DATERR= 000000           129#
DMACSR= 021002             9#    243     252*
DMAF   = 021022           17#
DMAINT   000472R   002   161     243#
DMAT0  = 021004           10#
DMAT1  = 021006           11#
DMAVEC= 000114            98#    162*    163*
DMAWC  = 021024           18#
DMDONE   000642R   002   243*    244     249     281#
DNI    = 004000           72#    146     224
DRTY   = 000040           88#
DTCR   = 000010           90#    496     638     736
ENABLE= 100000            78#    384     497     639     737     782     868     913
ENAL   = 040000           85#
ENCR   = 004000           86#
ERRFLG= 000002           105#    234
ERRINT   000446R   002   148     234#
EXCRCH= 157273           136#    412
EXCRCL= 020343           137#    414
FATI   = 000400           74#
FLG6     000624R   002   193*    196     200     225*    231*    234*    256*    264*    267*    276#    390*    395     398
                         503*    508     511     646*    651     654     744*    751     759     788*    793     796     875*
                         882     890     919*    924     927
GETCRC   000720R   002   302     311#
HDX    = 000001           92#    867
```

```
INERR = 000003            103#   222
INITH = 177777            132#   299
INITL = 177777            133#   300
INMON = 000001            101#   144    208    220
INTST = 000002            102#   341    435    545    691    835
IOADR = 020000            124#   125
IOSIZ = 020000            120#   125
IPCSR0= 021000              8#   146*   205    224*   235*   246*   251*   257*
IPCSR1= 021020             16#   145*   208*   223*   342*   436*   546*   692*   836*
IPCSR2  000632R     002   191*   192*   279#   348    349    443    444    524    525    553    554    587    588
                          667    668    722    723    813    814    944    945
LASFTP= 012400             56#
LFRBUF= 021032             21#   391*   504*   647*   745*   789*   876*   920*
LINADR= 100000            126#   361    364    374    456    459    469    521    566    569    579    701    704
                          714    772    845    848    858    903
LINSIZ= 077774            122#
LOOP  = 000004             91#   383    496    638    736    781    867    912
LRBUF = 021044             26#   262    806    937
LTAC  = 021030             20#   392*   505*   648*   746*   790*   877*   921*
MAXBC = 002752            128#   388    410    479    483    501
MDMAR0= 021014             14#   350    351    354    445    446    449    526    527    555    556    559    589
                          590    595    669    670    724    725    728    815    816    946    947
MDMAR1= 021016             15#
MDMAW0= 021026             19#   532*   675*   680*   822*   823*   951*   952*
MDMAW1= 021036             23#
MDMA0 = 021010             12#   348*   352*   443*   447*   524*   530*   553*   557*   587*   593*   667*   673*
                          722*   726*   813*   819*   944*   948*
MDMA1 = 021012             13#   349*   353*   444*   448*   525*   531*   554*   558*   588*   594*   668*   674*
                          723*   727*   814*   820*   945*   949*
MDMSR = 021042             25#
MICFSZ  003456RG    002   959#
MICF1   001002R     002   270    341#
MICF2   001274R     002   271    435#
MICF3   001650R     002   272    545#
MICF4   002364R     002   273    691#
MICF5   003044R     002   274    835#
MICROF  000000RG    002   141#   959
MINBC = 000100            127#   786    917
MODE  = 000200             79#   382    495    637    735    780    866    911
MODREG= 177774             33#   383*   496*   638*   736*   781*   867*   912*
NPRERR= 100000            111#   251
NPRFLG= 000020            108#
NXMERR= 040000            112#   246
NXMFLG= 000010            107#
PARERR= 010000            114#   257
PARFLG= 000004            106#   256
PARINT  000550R     002   156    256#
PARVEC= 000140             99#   157*   158*
PCBADR  000636R     002   280#
PCEI  = 040000             69#
PCSRSW= 021040             24#   185
PCSR1   000630R     002   144*   145    220*   222*   223    278#   341*   342    417*   435*   436    535*   545*
                          546    681*   691*   692    827*   835*   836    956*
PHYAD0= 021060             27#
PHYAD1= 021062             28#
PHYAD2= 021064             29#
```

```
PHYAD3= 021066            30#
PHYAD4= 021070            31#
PHYAD5= 021072            32#
POLYH = 166670           134#   317
POLYL = 101440           135#   318
PRI00 = 000000            64#   194*
PRI01 = 000040            63#
PRI02 = 000100            62#
PRI03 = 000140            61#   394*   507*   650*   748*   792*   879*   923*
PRI04 = 000200            60#   168    178
PRI05 = 000240            59#   173    183
PRI06 = 000300            58#
PRI07 = 000340            57#   141*   151    158    163    199*   401*   514*   657*   799*   930*
PROM  = 100000            84#   383    496    638    736    781    867    912
RBUFP   000644R   002    282#   362*   391    408    457*   504    567*   647    664    702*   745    789    808
                         846*   876    920    939
RCEI  = 002000            73#
RCVFLG= 000100           110#   264    395    508    651    751    793    882    924
RCVINT  000566R   002    181    262#
RCVVEC= 000120            95#   182*   183*
ROMADR= 040000           125#   126
ROMSIZ= 040000           121#   126
RXI   = 020000            70#
SANTIM  000626R   002    240*   277#   694*   749    838*   880
SANVEC= 000134            96#   172*   173*
SERI  = 100000            68#
SIZ1K = 004000           115#   116    364    459    569    704    848
SIZ2K = 010000           116#   117    374    469    579    714    772    858    903
SIZ4K = 020000           117#   118    119    120
SIZ8K = 040000           118#   121    122
STACK = 001000           100#   143
TBLD    000612R   002    214    270#
TBUFP   000646R   002    283#   372*   386    392    467*   477    481    499    505    577*   602    606    641
                         648    678    712*   739    746    770    784    790    856*   870    877    901    915
                         921
TIMINT  000464R   002    171    240#
TRNFLG= 000040           109#   267    398    511    654    759    796    890    927
TRNINT  000602R   002    176    267#
TRNVEC= 000070            94#   177*   178*
TXI   = 010000            71#
UNIERR= 020000           113#   235
WCSADR= 000000           123#   124
WCSSIZ= 020000           119#   124
.     = 003460R   002    148    156    161    166    171    176    181    202    214    238    253    259    270
                         271    272    273    274
```

```
BCOMPL      1#
BERROR      1#
BGNAU       1#
BGNAUT      1#
BGNCLN      1#
BGNDU       1#
BGNHRD      1#
BGNHW       1#
BGNINI      1#
BGNMOD      1#
BGNMSG      1#
BGNPRO      1#
BGNPTA      1#
BGNRPT      1#
BGNSEG      1#
BGNSET      1#
BGNSFT      1#
BGNSRV      1#
BGNSUB      1#
BGNSW       1#
BGNTST      1#
BNCOMP      1#
BNERRO      1#
BREAK       1#
BRESET      1#
CKLOOP      1#
CLOCK       1#
CLOSE       1#
CLRVEC      1#
COMMEN      1#
DELAY       1#
DESCRI      1#
DEVTYP      1#
DISPAT      1#
DISPLA      1#
DOCLN       1#
DODU        1#
DORPT       1#
ENDAU       1#
ENDAUT      1#
ENDCLN      1#
ENDCOM      1#
ENDDU       1#
ENDHRD      1#
ENDHW       1#
ENDINI      1#
ENDMOD      1#
ENDMSG      1#
ENDPRO      1#
ENDPTA      1#
ENDRPT      1#
ENDSEG      1#
ENDSET      1#
ENDSFT      1#
ENDSRV      1#
ENDSUB      1#
```

```
ENDSW       1#
ENDTST      1#
EQUALS      1#
ERRDF       1#
ERRHRD      1#
ERROR       1#
ERRSF       1#
ERRSOF      1#
ERRTBL      1#
ESCAPE      1#
EXIT        1#
FEQUAL      1#
GETBYT      1#
GETPRI      1#
GETWOR      1#
GMANIA      1#
GMANID      1#
GMANIL      1#
GPHARD      1#
GPRMA       1#
GPRMD       1#
GPRML       1#
HEADER      1#
INLOOP      1#
IOSETU      1#
IOSTAR      1#
KT11        1#
LASTAD      1#
MANUAL      1#
MEMORY      1#
MSBYTE      1#
MSCHEC      1#
MSCNTO      1#
MSCOUN      1#
MSDATA      1#
MSDECR      1#
MSDEFA      1#
MSENDE      1#
MSERRI      1#
MSESCA      1#
MSESCS      1#
MSEXCP      1#
MSEXIT      1#
MSEXSE      1#
MSEXTJ      1#
MSGEN       1#
MSGENB      1#
MSGETS      1#
MSGETT      1#
MSGNGB      1#
MSGNIN      1#
MSGNLS      1#
MSGNSU      1#
MSGNTA      1#
MSGNTE      1#
MSHAPT      1#
```

```
MSHNAP    1#
MSINCR    1#
MSIOSE    1#
MSLDRO    1#
MSMASK    1#
MSMCHI    1#
MSMCLO    1#
MSMSK1    1#
MSPOP     1#
MSPRIN    1#
MSPUSH    1#
MSPUT     1#
MSPUT1    1#
MSRADI    1#
MSRBRO    1#
MSRNRO    1#
MSSETS    1#
MSSTAR    1#
MSSVC     1#
MSTLAB    1#
MSTSTL    1#
MSWORD    1#
MSXFER    1#
OPEN      1#
POINTE    1#
PRINTB    1#
PRINTF    1#
PRINTS    1#
PRINTX    1#
READBU    1#
READEF    1#
RFLAGS    1#
SETPRI    1#
SETVEC    1#
SLASH     1#
STARS     1#
SVC       1#
XFER      1#
XFERF     1#
XFERT     1#

. ABS.  000000    000
        000000    001
MICRF   003460    002

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROF.OBJ,MICROF.LST/CR/SOL/NL:TOC=SVC34R.P11,MICROF.MAC
 RUN-TIME: 2 3 .4 SECONDS
 RUN-TIME RATIO: 40/6=6.0
 CORE USED:  31K  (61 PAGES)
```

```
                        .TITLE  MICROG - MICROCODE MODULE G

        000000'         .CSECT MICRG

                        .SBTTL   REGISTER DEFINITIONS USED BY THE T11

        021000          IPCSR0  =       21000   ;INTERNAL PCSR0 ADDRESS
        021002          DMACSR  =       21002   ;DMA ENGINE CONTROL STATUS REGISTER
        021004          DMAT0   =       21004   ;DMA ENGINE TO ADDRESS REGISTER #0
        021006          DMAT1   =       21006   ;DMA ENGINE TO ADDRESS REGISTER #1
        021010          MDMA0   =       21010   ;MICROCPU DMA TO ADDRESS REGISTER #0
        021012          MDMA1   =       21012   ;MICROCPU DMA TO ADDRESS REGISTER #1
        021014          MDMAR0  =       21014   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT R/O
        021016          MDMAR1  =       21016   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT R/O
        021020          IPCSR1  =       21020   ;INTERNAL PCSR1 ADDRESS
        021022          DMAF    =       21022   ;DMA ENGINE FROM ADDRESS REGISTER
        021024          DMAWC   =       21024   ;DMA ENGINE WORD COUNT REGISTER
        021026          MDMAW0  =       21026   ;MICROCPU DMA DATA REGISTER - AUTO INCREMENT W/O
        021030          LTAC    =       21030   ;LINK TRANSMIT ADDRESS COUNTER REGISTER
        021032          LFRBUF  =       21032   ;LINK RECIEVE BUFFER ADDRESS FIFO
        021034          CLRFIF  =       21034   ;CLEAR FIFO
        021036          MDMAW1  =       21036   ;MICROCPU DMA DATA REGISTER - AUTO DECREMENT W/O
        021040          PCSRSW  =       21040   ;SWITCH PACK REGISTER
        021042          MDMSR   =       21042   ;MICROCPU DMA STATUS REGISTER
        021044          LRBUF   =       21044   ;LINK RECIEVE BUFFER COMPLETED
        021060          PHYAD0  =       21060   ;PHYSICAL ADDRESS ROM BYTE 0
        021062          PHYAD1  =       21062   ;PHYSICAL ADDRESS ROM BYTE 1
        021064          PHYAD2  =       21064   ;PHYSCIAL ADDRESS ROM BYTE 2
        021066          PHYAD3  =       21066   ;PHYSICAL ADDRESS ROM BYTE 3
        021070          PHYAD4  =       21070   ;PHYSICAL ADDRESS ROM BYTE 4
        021072          PHYAD5  =       21072   ;PHYSICAL ADDRESS ROM BYTE 5
        177774          MODREG  =       177774  ;LINK MODE REGISTER
        177774          ADRREG  =       177774  ;LINK STATION ADDRESS RAM REGISTER
        177776          CMDREG  =       177776  ;LINK COMMAND REGISTER

                        .SBTTL   OTHER DEFINITIONS USED BY THE MICROCODE

        100000          BIT15   =       100000
        040000          BIT14   =       40000
        020000          BIT13   =       20000
        010000          BIT12   =       10000
        004000          BIT11   =       4000
        002000          BIT10   =       2000
        001000          BIT9    =       1000
        000400          BIT8    =       400
        000200          BIT7    =       200
        000100          BIT6    =       100
        000040          BIT5    =       40
        000020          BIT4    =       20
        000010          BIT3    =       10
        000004          BIT2    =       4
        000002          BIT1    =       2
        000001          BIT0    =       1
                        ;
        012400          LASFTP  =       BIT8!BIT10!BIT12 ;LOAD AND START FUNCTION TEST PATTERN
```

```
        000340          PRI07   =       340
        000300          PRI06   =       300
        000240          PRI05   =       240
        000200          PRI04   =       200
        000140          PRI03   =       140
        000100          PRI02   =       100
        000040          PRI01   =       40
        000000          PRI00   =       0
                        ;
                        ;PCSR0 - PORT CONTROL STATUS REGISTER 0
                        ;
        100000          SERI    =       BIT15
        040000          PCEI    =       BIT14
        020000          RXI     =       BIT13
        010000          TXI     =       BIT12
        004000          DNI     =       BIT11
        002000          RCEI    =       BIT10
        000400          FATI    =       BIT8
                        ;
                        ;LINK COMMAND REGISTER
                        ;
        100000          ENABLE  =       BIT15   ;ENABLE LINK MODULE
        000200          MODE    =       BIT7    ;ENABLE MODE REGISTER
        000100          ARAM    =       BIT6    ;ENABLE STATION ADDRESS RAM
                        ;
                        ;LINK MODE REGISTER
                        ;
        100000          PROM    =       BIT15   ;PROMISCUIOUS MODE
        040000          ENAL    =       BIT14   ;ENABLE MULTICAST
        004000          ENCR    =       BIT11   ;ENABLE COLLISION TEST
        002000          ACLO    =       BIT10   ;ENABLE ACLO
        000040          DRTY    =       BIT5    ;DISABLE RETRY LOGIC
        000020          COLL    =       BIT4    ;SIMULATE A COLLISION
        000010          DTCR    =       BIT3    ;DISABLE TRANSMIT CRC LOGIC
        000004          LOOP    =       BIT2    ;ENABLE LOOPBACK

        000070          TRNVEC= 70              ;VECTOR ADDRESS FOR THE TRANSMITTER
        000120          RCVVEC= 120             ;VECTOR ADDRESS FOR THE RECEIVER
        000134          SANVEC= 134             ;VECTOR ADDRESS FOR THE SANITY TIMER
        000064          CSRVEC= 64              ;VECTOR ADDRESS FOR CSR WRITE INTERRUPT
        000114          DMAVEC= 114             ;VECTOR ADDRESS FOR DMA DONE INTERRUPT
        000140          PARVEC= 140             ;VECTOR ADDRESS FOR LINK MEMORY PARITY ERROR
        001000          STACK=  1000            ;STACK LOCATION
        000001          INMON=  1               ;IN MICROMONITOR STATE
        000002          INTST=  2               ;IN A TEST STATE
        000003          INERR=  3               ;IN ERROR STATE
        000001          CSRFLG= BIT0            ;CSR WRITE INTERRUPT OCCURED
        000002          ERRFLG= BIT1            ;UNEXPECTED ERROR OCCURED
        000004          PARFLG= BIT2            ;PARITY ERROR OCCURED
        000010          NXMFLG= BIT3            ;NON-EXISTANT MEMORY ERROR OCCURRED
        000020          NPRFLG= BIT4            ;NPR TIMEOUT OCCURRED
        000040          TRNFLG= BIT5            ;TRANSMITTER INTERRUPT OCCURRED
        000100          RCVFLG= BIT6            ;RECEIVER INTERRUPT OCCURRED
        100000          NPRERR= BIT15           ;PCSR0 FLAG INDICATING NPR ERROR OCCURRED
        040000          NXMERR= BIT14           ;PCSR0 FLAG INDICATING NON-EXISTANT MEMORY ERROR OCCURRED
        020000          UNIERR= BIT13           ;PCSR0 FLAG INDICATING UNEXPECTED INTERRUPT OCCURRED
```

```
010000     PARERR= BIT12            ;PCSRO FLAG INDICATING LINK MEMORY PARITY ERROR OCCURRED
004000     SIZ1K=  4000             ;1K WORDS
010000     SIZ2K=  SIZ1K*2          ;2K WORDS
014000     SIZ3K=  SIZ1K*3          ;3K WORDS
020000     SIZ4K=  SIZ2K*2          ;4K WORDS
040000     SIZ8K=  SIZ4K*2          ;8K WORDS
020000     WCSSIZ= SIZ4K            ;SIZE OF WRITEABLE CONTROL STORE
020000     IOSIZ=  SIZ4K            ;SIZE OF I/O PAGE
040000     ROMSIZ= SIZ8K            ;SIZE OF ROM
077774     LINSIZ= SIZ8K*2-4        ;SIZE OF LINK MEMORY
000000     WCSADR= 0                ;BASE ADDRESS OF WCS
020000     IOADR=  WCSADR+WCSSIZ    ;BASE ADDRESS OF I/O PAGE
040000     ROMADR= IOADR+IOSIZ      ;BASE ADDRESS OF ROM
100000     LINADR= ROMADR+ROMSIZ    ;BASE ADDRESS OF LINK MEMORY
002756     MAXBC=  1518.            ;MAXIMUM # OF BYTES IN A TRANSMIT BUFFER
000000     DATERR= 0                ;FLAG INDICATING DATA ERROR OCCURRED
000004     CRCSIZ= 4                ;NUMBER OF BYTES IN A CRC
000001     ADRERR= 1                ;FLAG INDICATING ADDRESS ERROR OCCURRED
```

```

                                          .SBTTL  G_MODULE MICROCODE

000000' 106427  000340            MICROG::MTPS    #PRI07                 ;DISABLE INTERRUPTS
000004' 012706  001000                    MOV     #STACK,SP              ;SETUP STACK
000010' 112767  000001  000600            MOVB    #INMON,PCSR1           ;TELL HOST WE ARE IN MICROMONITOR
000016' 016737  000574  021020            MOV     PCSR1,@#IPCSR1
000024' 012737  004000  021000            MOV     #DNI,@#IPCSR0          ;TELL HOST THE LOAD AND START FINISHED
000032' 010700                            MOV     PC,R0                  ;GET ADDRESS OF UNEXPECTED ERROR...
000034' 062700  000404                    ADD     #ERRINT-.,R0           ;HANDLER
000040' 005001                            CLR     R1                     ;FILL ALL UNUSED VECTORS WITH TRAP...
000042' 010021                    10$:    MOV     R0,(R1)+               ;HANDLER
000044' 012721  000340                    MOV     #PRI07,(R1)+
000050' 020127  001000                    CMP     R1,#1000
000054' 002772                            BLT     10$

000056' 010700                            MOV     PC,R0                  ;SETUP PARITY TRAP VECTOR
000060' 062700  000462                    ADD     #PARINT-.,R0
000064' 010037  000140                    MOV     R0,@#PARVEC
000070' 012737  000340  000142            MOV     #PRI07,@#PARVEC+2

000076' 010700                            MOV     PC,R0                  ;SETUP DMA INTERRUPT VECTOR
000100' 062700  000364                    ADD     #DMAINT-.,R0
000104' 010037  000114                    MOV     R0,@#DMAVEC
000110' 012737  000340  000116            MOV     #PRI07,@#DMAVEC+2

000116' 010700                            MOV     PC,R0                  ;SETUP CSR WRITE VECTOR
000120' 062700  000310                    ADD     #CSRWRT-.,R0
000124' 010037  000064                    MOV     R0,@#CSRVEC
000130' 012737  000200  000066            MOV     #PRI04,@#CSRVEC+2

000136' 010700                            MOV     PC,R0                  ;SETUP SANTITY TIMER VECTOR
000140' 062700  000316                    ADD     #TIMINT-.,R0
000144' 010037  000134                    MOV     R0,@#SANVEC
000150' 012737  000240  000136            MOV     #PRI05,@#SANVEC+2

000156' 010700                            MOV     PC,R0                  ;SETUP TRANSMITTER VECTOR
000160' 062700  000414                    ADD     #TRNINT-.,R0
000164' 010037  000070                    MOV     R0,@#TRNVEC
000170' 012737  000200  000072            MOV     #PRI04,@#TRNVEC+2

000176' 010700                            MOV     PC,R0                  ;SETUP RECEIVER VECTOR
000200' 062700  000360                    ADD     #RCVINT-.,R0
000204' 010037  000120                    MOV     R0,@#RCVVEC
000210' 012737  000240  000122            MOV     #PRI05,@#RCVVEC+2

000216' 013700  021040                    MOV     @#PCSRSW,R0            ;GET SWITCH PACK BITS
000222' 052700  176000                    BIS     #176000,R0             ;MAP THEM INTO HOST I/O PAGE
000226' 006300                            ASL     R0                     ;SHIFT OVER TO POSITION CORRECTLY
000230' 006300                            ASL     R0
000232' 006300                            ASL     R0
000234' 062700  000004                    ADD     #4,R0                  ;PCSR2 IS PCSR0+4
000240' 010067  000354                    MOV     R0,IPCSR2              ;SAVE PCSR2 ADDRESS
000244' 012767  000003  000350            MOV     #3,IPCSR2+2            ;HIGH ORDER BITS 17:16
000252' 005067  000334                    CLR     FLG7                   ;INITIALIZE FLAG WORD
000256' 106427  000000            15$:    MTPS    #PRI00                 ;ALLOW INTERRUPTS
```

```
000262' 005767 000324          20$:    TST     FLG7                    ;WAIT FOR A COMMAND FROM HOST
000266' 001775                         BEQ     20$

000270' 106427 000340                  MTPS    #PR↓07                  ;RAISE CPU PRIORITY TO SERVICE COMMAND
000274' 032767 000001 000310           BIT     #CSRFLG,FLG7            ;DID HOST GIVE US A COMMAND?
000302' 001001                         BNE     30$                     ;YES
000304' 000777                         BR      .                       ;NO, ERROR SO JUST SIT HERE...
                                                                       ;FOR LACK OF ANYTHING BETTER TO DO

000306' 113700 021000          30$:    MOVB    @#IPCSR0,R0             ;GET WHAT HOST WROTE TO PCSR0
000312' 042700 177760                  BIC     #177760,R0              ;STRIP ALL BUT COMMAND BITS
000316' 001004                         BNE     35$                     ;WAS IT THE CLEAR FUNCTION?
000320' 012737 000001 021020           MOV     #INMON,@#IPCSR1         ;YES, CLEAR OUT THE TEST # BITS
000326' 000432                         BR      50$
000330' 022700 000017          35$:    CMP     #17,R0                  ;RESTART OPERATIONAL MICROCODE?
000334' 001432                         BEQ     60$                     ;YES
000336' 162700 000001                  SUB     #1,R0
000342' 010701                         MOV     PC,R1                   ;GET ADDRESS OF OUR COMMAND TABLE
000344' 062701 000240                  ADD     #TBLG-.,R1
000350' 006300                         ASL     R0                      ;MAKE COMMAND A BYTE OFFSET
000352' 060001                         ADD     R0,R1                   ;USE IT TO INDEX INTO COMMAND TABLE
000354' 061101                         ADD     (R1),R1                 ;R1 NOW HAS COMMAND ROUTINE ADDRESS
000356' 004711                         JSR     PC,(R1)                 ;EXECUTE AS COMMANDED FROM HOST
000360' 103404                         BCS     40$                     ;ERROR OCCURRED
000362' 112767 000001 000226           MOVB    #INMON,PCSR1            ;INDICATE TO HOST WE ARE BACK IN...
000370' 000403                         BR      45$                     ;M↓CROMONITR
000372' 112767 000003 000216   40$:    MOVB    #INERR,PCSR1            ;INDICATE TO HOST ERROR OCCURRED
000400' 016737 000212 021020   45$:    MOV     PCSR1,@#IPCSR1
000406' 012737 004000 021000           MOV     #DNI,@#IPCSR0           ;TELL HOST THIS MICROTEST FINISHED
000414' 005067 000172          50$:    CLR     FLG7                    ;RESET FLAG WORD
000420' 000716                         BR      15$                     ;GO WAIT FOR ANOTHER COMMAND

000422' 005000                 60$:    CLR     R0                      ;FAKE SUCCESSFUL SELF TEST RESULTS
000424' 000137 040006                  JMP     @#40006                 ;START OPERATIONAL MICROCODE

000430' 052767 000001 000154   CSRWRT: BIS     #CSRFLG,FLG7            ;INDICATE A CSR WRITE INTERRUPT OCCURED
000436' 000002                         RTI

000440' 052767 000002 000144   ERRINT: BIS     #ERRFLG,FLG7            ;INDICATE A UNEXPECTED INTERRUPT OCCURED
000446' 012737 020000 021000           MOV     #UNIERR,@#IPCSR0        ;TELL HOST AN UNEXPECTED INTERRUPT
                                                                       ;HAPPENED
                                                                       ;JUST SIT HERE AND SPIN WHEELS
000454' 000777                         BR      .                       ;COUNT ON THE HOST TO TIMEOUT WAITING


000456' 005267 000132          TIMINT: INC     SANTIM                  ;COUNT TICKS AS THEY OCCUR
000462' 000002                         RTI


000464' 013767 021002 000136   DMAINT: MOV     @#DMACSR,DMDONE         ;GET DMA STATUS
000472' 032767 040000 000130           BIT     #BIT14,DMDONE           ;DID A NON-EXISTANT MEMORY INTERRUPT OCCUR?
000500' 001404                         BEQ     10$                     ;NO
000502' 012737 040000 021000           MOV     #NXMERR,@#IPCSR0        ;YES, TELL HOST A NON-EXISTANT MEMORY
                                                                       ;LOCATION WAS ADDRESSED
000510' 000407                         BR      20$
000512' 032767 100000 000110   10$:    BIT     #BIT15,DMDONE           ;DID A NPR TIMEOUT OCCUR?
```

```
000520' 001407                          BEQ     30$                     ;NO
000522' 012737  100000  021000          MOV     #NPRERR,@#IPCSR0        ;TELL HOST NPR TIMEOUT HAPPENED
000530' 012737  100000  021002  20$:    MOV     #BIT15,@#DMACSR         ;CLEAR THE INTERRUPT IN THE DMA ENGINE
000536' 000777                          BR      .                       ;SIT HERE AND SPIN WHEELS
000540' 000002                  30$:    RTI

000542' 052767  000004  000042  PARINT: BIS     #PARFLG,FLG7            ;SET PARITY ERROR OCCURRED
000550' 012737  010000  021000          MOV     #PARERR,@#IPCSR0        ;TELL HOST A LINK MEMORY PARITY ERROR
                                                                        ;OCCURRED
000556' 000777                          BR      .                       ;JUST SIT HERE AND SPIN WHEELS
                                                                        ;COUNT ON HOST TO TIMEOUT

000560' 005737  021044          RCVINT: TST     @#LRBUF                 ;READ BUFFER DONE REGISTER...
                                                                        ;WHICH CLEARS THE INTERRUPT
000564' 052767  000100  000020          BIS     #RCVFLG,FLG7            ;SET RECEIVER INTERRUPT OCCURRED
000572' 000002                          RTI

000574' 052767  000040  000010  TRNINT: BIS     #TRNFLG,FLG7            ;SET TRANSMITTER INTERRUPT OCCURRED
000602' 000002                          RTI

000604' 000036                  TBLG:   .WORD   MICG1-.                 ;COLLISION TEST
000606' 000306                          .WORD   MICG2-.                 ;TDR COUNTER TEST
000610' 000556                          .WORD   MICG3-.                 ;RETRY TEST

000612' 000000                  FLG7:   .WORD   0                       ;FLAG WORD
000614' 000000                  SANTIM: .WORD   0                       ;COUNT FOR SANITY TIMER
000616' 000000                  PCSR1:  .WORD   0                       ;CUPY OF WHAT GOES TO PCSR1
000620' 000000  000000          IPCSR2: .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCSR2
000624' 000000  000000          PCBADR: .WORD   0,0                     ;ADDRESS IN HOST MEMORY FOR PCB
000630' 000000                  DMDONE: .WORD   0
000632' 000000                  RBUFP:  .WORD   0                       ;POINTER TO RECIEVE BUFFER
000634' 000000                  TBUFP:  .WORD   0                       ;POINTER TO XMIT BUFFER
000636' 000000                  DBUFP:  .WORD   0                       ;POINTER TO DMA ENGINE BUFFER
000640' 000000                  MBUFP:  .WORD   0                       ;POINTER TO MICROCPU DMA BUFFER
```

```
000642' 112767  000002  177746  MICG1:  MOVB    #INTST,PCSR1            ;TELL HOST WE ARE IN A TEST
000650' 016737  177742  021020          MOV     PCSR1,@#IPCSR1

000656' 012700  100000                  MOV     #LINADR,R0              ;FILL RECEIVE BUFFER WITH ZEROS
000662' 010067  177744                  MOV     R0,RBUFP
000666' 005020                  10$:    CLR     (R0)+
000670' 020027  104000                  CMP     R0,#LINADR+SIZ1K
000674' 103774                          BLO     10$

000676' 010067  177732                  MOV     R0,TBUFP                ;FILL TRANSMIT BUFFER WITH 1'S
000702' 012720  177777          20$:    MOV     #177777,(R0)+
000706' 020027  110000                  CMP     R0,#LINADR+SIZ2K
000712' 103773                          BLO     20$

000714' 016737  177700  021010          MOV     IPCSR2,@#MDMA0          ;GET HOST'S PCBB ADDRESS
000722' 016737  177674  021012          MOV     IPCSR2+2,@#MDMA1
000730' 013704  021014                  MOV     @#MDMAR0,R4             ;R4 HOLDS PCBB LOW ADDRESS
000734' 013705  021014                  MOV     @#MDMAR0,R5             ;R5 HOLDS PCBB HIGH ADDRESS
000740' 010437  021010                  MOV     R4,@#MDMA0              ;POINT TO PCBB+0
000744' 010537  021012                  MOV     R5,@#MDMA1
000750' 013703  021014                  MOV     @#MDMAR0,R3             ;GET WHAT HOST WANTS TO GO INTO LINK
                                                                        ;MODE REGISTER

000754' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ;ENABLE LINK MODULE AND SELECT MODE REG
000762' 010337  177774                  MOV     R3,@#MODREG             ;LOAD MODE REGISTER WITH HOST'S VALUE

000766' 016702  177642                  MOV     TBUFP,R2                ;GET BEGINNING OF TRANSMIT BUFFER
000772' 005722                          TST     (R2)+                   ;SKIP FIRST WORD
000774' 012722  002752                  MOV     #MAXBC-CRCSIZ,(R2)+     ;SET TO TRANSMIT MAX LENGTH PACKET
001000' 005037  021034                  CLR     @#CLRFIF                ;CLEAR THE FIFO
001004' 005067  177602                  CLR     FLG7                    ;CLEAR OUT THE FLAG WORD
001010' 016737  177616  021032          MOV     RBUFP,@#LFRBUF          ;TELL UNA WHERE RECEIVE BUFF IS
001016' 016737  177612  021030          MOV     TBUFP,@#LTAC            ;TELL UNA WHERE TRANSMIT BUFF IS
                                                                        ;START TRANSMIT OPERATION

001024' 106427  000140                  MTPS    #PRI03                  ;ALLOW INTERRUPTS
001030' 032767  000040  177554  30$:    BIT     #TRNFLG,FLG7            ;WAIT FOR TRANSMIT DONE
001036' 001774                          BEQ     30$

001040' 106427  000340                  MTPS    #PRI07                  ;DISABLE ANY MORE INTERRUPTS

001044' 016700  177564                  MOV     TBUFP,R0                ;POINT TO TRANSMIT BUFFER
001050' 012001                          MOV     (R0)+,R1                ;GET FIRST TRANSMIT STATUS WORD
001052' 012002                          MOV     (R0)+,R2                ;GET SECOND TRANSMIT STATUS WORD

001054' 062704  000002                  ADD     #2,R4                   ;POINT TO HOSTS PCBB+2
001060' 005505                          ADC     R5
001062' 010437  021010                  MOV     R4,@#MDMA0
001066' 010537  021012                  MOV     R5,@#MDMA1
001072' 010137  021026                  MOV     R1,@#MDMAW0             ;DUMP FIRST STATUS WORD TO PCBB+2
001076' 010237  021026                  MOV     R2,@#MDMAW0             ;DUMP SECOND STATUS WORD TO PCBB+4

001102' 112767  000001  177507          MOVB    #1,PCSR1+1              ;TELL HOST WHAT TEST JUST FINISHED
001110' 000241                          CLC
001112' 000207                          RTS     PC
```

```
001114' 112767  000002  177474  MICG2:  MOVB    #INTST,PCSR1            ;TELL HOST THAT WE ARE IN A TEST
001122' 016737  177470  021020          MOV     PCSR1,@#IPCSR1

001130' 012700  100000                  MOV     #LINADR,R0              ;FILL RECEIVE BUFFER WITH ALL 1'S
001134' 010067  177472                  MOV     R0,RBUFP
001140' 012720  177777          10$:    MOV     #177777,(R0)+
001144' 020027  104000                  CMP     R0,#LINADR+SIZ1K
001150' 103773                          BLO     10$

001152' 010067  177456                  MOV     R0,TBUFP                ;FILL TRANSMIT BUFFER WITH ALL 0'S
001156' 005020                  20$:    CLR     (R0)+
001160' 020027  110000                  CMP     R0,#LINADR+SIZ2K
001164' 103774                          BLO     20$

001166' 016737  177426  021010          MOV     IPCSR2,@#MDMA0          ;GET HOST'S PCBB ADDRESS
001174' 016737  177422  021012          MOV     IPCSR2+2,@#MDMA1
001202' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 HOLDS HOST'S PCBB LOW ADDRESS
001206' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 HOLDS HOST'S PCBB HI ADDRESS
001212' 010037  021010                  MOV     R0,@#MDMA0              ;POINT TO PCBB+0
001216' 010137  021012                  MOV     R1,@#MDMA1
001222' 013703  021014                  MOV     @#MDMAR0,R3             ;GET BYTE COUNT FOR TRANSMIT OPERATION

001226' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ;ENABLE LINK MODULE AND SELECT MODE REG
001234' 012737  100004  177774          MOV     #PROM!LOOP,@#MODREG     ;ACCEPT ALL PACKETS IN LOOP BACK

001242' 016704  177366                  MOV     TBUFP,R4                ;POINT TO TRANSMIT BUFFER
001246' 005724                          TST     (R4)+                   ;SKIP FIRST WORD
001250' 010324                          MOV     R3,(R4)+                ;SET BYTE COUNT TO HOST'S VALUE
001252' 005037  021034                  CLR     @#CLRFIF                ;CLEAR THE FIFO
001256' 005067  177330                  CLR     FLG7                    ;CLEAR THE FLAG WORD
001262' 016737  177344  021032          MOV     RBUFP,@#LFRBUF          ;TELL UNA WHERE RECEIVE BUFF IS
001270' 016737  177340  021030          MOV     TBUFP,@#LTAC            ;TELL UNA WHERE TRANSMIT BUFF IS AND
                                                                        ;START THE TRANSMIT OPERATION

001276' 106427  000140                  MTPS    #PRI03                  ;ALLOW INTERRUPTS
001302' 032767  000040  177302  30$:    BIT     #TRNFLG,FLG7            ;WAIT FOR TRANSMIT DONE INTERRUPT
001310' 001774                          BEQ     30$

001312' 106427  000340                  MTPS    #PRI07                  ;DISABLE INTERRUPTS

001316' 016702  177312                  MOV     TBUFP,R2                ;POINT TO TRANSMIT BUFFER
001322' 005722                          TST     (R2)+                   ;SKIP FIRST STATUS WORD
001324' 012203                          MOV     (R2)+,R3                ;GET SECOND STATUS WORD
001326' 042703  176000                  BIC     #176000,R3              ;STRIP ALL BUT TDR VALUE BITS
001332' 062700  000002                  ADD     #2,R0                   ;POINT TO HOST'S PCBB+2
001336' 005501                          ADC     R1
001340' 010037  021010                  MOV     R0,@#MDMA0
001344' 010137  021012                  MOV     R1,@#MDMA1
001350' 010337  021026                  MOV     R3,@#MDMAW0             ;DUMP TDR COUNTER VALUE TO PCBB+2

001354' 112767  000002  177235          MOVB    #2,PCSR1+1              ;TELL HOST WHAT TEST JUST FINISHED
001362' 000241                          CLC
001364' 000207                          RTS     PC

001366' 112767  000002  177222  MICG3:  MOVB    #INTST,PCSR1            ;TELL HOST THAT WE ARE IN A TEST
001374' 016737  177216  021020          MOV     PCSR1,@#IPCSR1
```

```
001402' 012700  100000                  MOV     #LINADR,R0              ;FILL RECEIVE BUFFER WITH ALL 1'S
001406' 010067  177220                  MOV     R0,RBUFP
001412' 012720  177777          10$:    MOV     #177777,(R0)+
001416' 020027  104000                  CMP     R0,#LINADR+SIZ1K
001422' 103773                          BLO     10$

001424' 010067  177204                  MOV     R0,TBUFP                ;FILL TRANSMIT BUFFER WITH ALL 0'S
001430' 005020                  20$:    CLR     (R0)+
001432' 020027  110000                  CMP     R0,#LINADR+SIZ2K
001436' 103774                          BLO     20$

001440' 016737  177154  021010          MOV     IPCSR2,@#MDMA0          ;GET HOST'S PCBB ADDRESS
001446' 016737  177150  021012          MOV     IPCSR2+2,@#MDMA1
001454' 013700  021014                  MOV     @#MDMAR0,R0             ;R0 HOLDS HOST'S PCBB LOW ADDRESS
001460' 013701  021014                  MOV     @#MDMAR0,R1             ;R1 HOLDS HOST'S PCBB HI ADDRESS
001464' 010037  021010                  MOV     R0,@#MDMA0              ;POINT TO PCBB+0
001470' 010137  021012                  MOV     R1,@#MDMA1
001474' 013703  021014                  MOV     @#MDMAR0,R3             ;GET BYTE COUNT FOR TRANSMIT OPERATION

001500' 012737  100200  177776          MOV     #MODE!ENABLE,@#CMDREG   ;ENABLE LINK MODULE AND SELECT MODE REG
001506' 012737  100024  177774          MOV     #PROM!LOOP!COLL,@#MODREG ;ACCEPT ALL PACKETS IN LOOP BACK AND
                                                                        ;FORCE A COLLISION


001514' 016704  177114                  MOV     TBUFP,R4                ;POINT TO TRANSMIT BUFFER
001520' 005724                          TST     (R4)+                   ;SKIP FIRST WORD
001522' 010324                          MOV     R3,(R4)+                ;SET BYTE COUNT TO HOST'S VALUE
001524' 005037  021034                  CLR     @#CLRFIF                ;CLEAR THE FIFO
001530' 005067  177056                  CLR     FLG7                    ;CLEAR THE FLAG WORD
001534' 016737  177072  021032          MOV     RBUFP,@#LFRBUF          ;TELL UNA WHERE RECEIVE BUFF IS
001542' 016737  177066  021030          MOV     TBUFP,@#LTAC            ;TELL UNA WHERE TRANSMIT BUFF IS AND
                                                                        ;START THE TRANSMIT OPERATION


001550' 005005                          CLR     R5                      ;CLEAR COUNTER
001552' 106427  000140                  MTPS    #PRI03                  ;ALLOW INTERRUPTS
001556' 032767  000040  177026  30$:    BIT     #TRNFLG,FLG7            ;IS TRANSMITTER DONE?
001564' 001002                          BNE     35$                     ;YES
001566' 005205                          INC     R5                      ;NO, COUNT TIME
001570' 000772                          BR      30$
001572' 106427  000340          35$:    MTPS    #PRI07                  ;DISABLE INTERRUPTS

001576' 062700  000002                  ADD     #2,R0                   ;POINT TO HOST'S PCBB+2
001602' 005501                          ADC     R1
001604' 010037  021010                  MOV     R0,@#MDMA0
001610' 010137  021012                  MOV     R1,@#MDMA1
001614' 010537  021026                  MOV     R5,@#MDMAW0             ;DUMP COUNTER VALUE TO PCBB+2

001620' 112767  000003  176771          MOVB    #3,PCSR1+1              ;TELL HOST WHAT TEST JUST FINISHED
001626' 000241                          CLC
001630' 000207                          RTS     PC

001632' 001634                  MICGSZ::MICGSZ-MICROG+2
        000001                  .END
```

```
ACLO  = 002000          87#
ADRERR= 000001         130#
ADRREG= 177774          34#
ARAM  = 000100          80#
BIT0  = 000001          54#    103
BIT1  = 000002          53#    104
BIT10 = 002000          44#     56      73      87
BIT11 = 004000          43#     72      86
BIT12 = 010000          42#     56      71     113
BIT13 = 020000          41#     70     112
BIT14 = 040000          40#     69      85     111     238
BIT15 = 100000          39#     68      78      84     110     243     246
BIT2  = 000004          52#     91     105
BIT3  = 000010          51#     90     106
BIT4  = 000020          50#     89     107
BIT5  = 000040          49#     88     108
BIT6  = 000100          48#     80     109
BIT7  = 000200          47#     79
BIT8  = 000400          46#     56      74
BIT9  = 001000          45#
CLRFIF= 021034          22#    308*    363*    418*
CMDREG= 177776          35#    302*    357*    411*
COLL  = 000020          89#    412
CRCSIZ= 000004         129#    307
CSRFLG= 000001         103#    193     224
CSRVEC= 000064          96#    160*    161*
CSRWRT  000430R   002  159     224#
DATERR= 000000         128#
DBUFP   000636R   002  276#
DMACSR= 021002           9#    237     246*
DMAF  = 021022          17#
DMAINT  000464R   002  154     237#
DMAT0 = 021004          10#
DMAT1 = 021006          11#
DMAVEC= 000114          97#    155*    156*
DMAWC = 021024          18#
DMDONE  000630R   002  237*    238     243     273#
DNI   = 004000          72#    139     217
DRTY  = 000040          88#
DTCR  = 000010          90#
ENABLE= 100000          78#    302     357     411
ENAL  = 040000          85#
ENCR  = 004000          86#
ERRFLG= 000002         104#    227
ERRINT  000440R   002  141     227#
FATI  = 000400          74#
FLG7    000612R   002  186*    189     193     218*    224*    227*    250*    258*    261*    268#    309*    315     364*
                       370     419*    426
INERR = 000003         102#    215
INMON = 000001         100#    137     201     213
INTST = 000002         101#    279     335     389
IOADR = 020000         124#    125
IOSIZ = 020000         120#    125
IPCSR0= 021000           8#    139*    198     217*    228*    240*    245*    251*
IPCSR1= 021020          16#    138*    201*    216*    280*    336*    390*
IPCSR2  000620R   002  184*    185*    271#    293     294     349     350     403     404
```

```
LASFTP= 012400              56#
LFRBUF= 021032              21#    310*   365*   420*
LINADR= 100000             126#    282    285    290    338    341    346    392    395    400
LINSIZ= 077774             122#
LOOP  = 000004              91#    358    412
LRBUF = 021044              26#    256
LTAC  = 021030              20#    311*   366*   421*
MAXBC = 002756             127#    307
MBUFP   000640R    002     277#
MDMAR0= 021014              14#    295    296    299    351    352    355    405    406    409
MDMAR1= 021016              15#
MDMAW0= 021026              19#    328*   329*   383*   436*
MDMAW1= 021036              23#
MDMA0 = 021010              12#    293*   297*   326*   349*   353*   381*   403*   407*   434*
MDMA1 = 021012              13#    294*   298*   327*   350*   354*   382*   404*   408*   435*
MDMSR = 021042              25#
MICGSZ  001632RG   002     442#
MICG1   000642R    002     264     279#
MICG2   001114R    002     265     335#
MICG3   001366R    002     266     389#
MICROG  000000RG   002     135#    442
MODE  = 000200              79#    302    357    411
MODREG= 177774              33#    303*   358*   412*
NPRERR= 100000             110#    245
NPRFLG= 000020             107#
NXMERR= 040000             111#    240
NXMFLG= 000010             106#
PARERR= 010000             113#    251
PARFLG= 000004             105#    250
PARINT  000542R    002     149     250#
PARVEC= 000140              98#    150*   151*
PCBADR  000624R    002     272#
PCEI  = 040000              69#
PCSRSW= 021040              24#    178
PCSR1   000616R    002     137*    138    213*   215*   216    270#   279*   280    331*   335*   336    385*   389*
                           390     438*
PHYAD0= 021060              27#
PHYAD1= 021062              28#
PHYAD2= 021064              29#
PHYAD3= 021066              30#
PHYAD4= 021070              31#
PHYAD5= 021072              32#
PRI00 = 000000              64#    187*
PRI01 = 000040              63#
PRI02 = 000100              62#
PRI03 = 000140              61#    314*   369*   425*
PRI04 = 000200              60#    161    171
PRI05 = 000240              59#    166    176
PRI06 = 000300              58#
PRI07 = 000340              57#    135*   144    151    156    192*   318*   373*   430*
PROM  = 100000              84#    358    412
RBUFP   000632R    002     274#    283*   310    339*   365    393*   420
RCEI  = 002000              73#
RCVFLG= 000100             109#    258
RCVINT  000560R    002     174     256#
RCVVEC= 000120              94#    175*   176*
```

```
ROMADR= 040000            125#   126
ROMSIZ= 040000            121#   126
RXI   = 020000             70#
SANTIM  000614R     002   233*   269#
SANVEC= 000134             95#   165*   166*
SERI  = 100000             68#
SIZ1K = 004000            114#   115    116    285    341    395
SIZ2K = 010000            115#   117    290    346    400
SIZ3K = 014000            116#
SIZ4K = 020000            117#   118    119    120
SIZ8K = 040000            118#   121    122
STACK = 001000             99#   136
TBLG    000604R     002   207    264#
TBUFP   000634R     002   275#   288*   305    311    320    344*   360    366    375    398*   415    421
TIMINT  000456R     002   164    233#
TRNFLG= 000040            108#   261    315    370    426
TRNINT  000574R     002   169    261#
TRNVEC= 000070             93#   170*   171*
TXI   = 010000             71#
UNIERR= 020000            112#   228
WCSADR= 000000            123#   124
WCSSIZ= 020000            119#   124
.     = 001634R     002   141    149    154    159    164    169    174    195    207    230    247    253    264
                          265    266
```

```
BCOMPL    1#
BERROR    1#
BGNAU     1#
BGNAUT    1#
BGNCLN    1#
BGNDU     1#
BGNHRD    1#
BGNHW     1#
BGNINI    1#
BGNMOD    1#
BGNMSG    1#
BGNPRO    1#
BGNPTA    1#
BGNRPT    1#
BGNSEG    1#
BGNSET    1#
BGNSFT    1#
BGNSRV    1#
BGNSUB    1#
BGNSW     1#
BGNTST    1#
BNCOMP    1#
BNERRO    1#
BREAK     1#
BRESET    1#
CKLOOP    1#
CLOCK     1#
CLOSE     1#
CLRVEC    1#
COMMEN    1#
DELAY     1#
DESCRI    1#
DEVTYP    1#
DISPAT    1#
DISPLA    1#
DOCLN     1#
DODU      1#
DORPT     1#
ENDAU     1#
ENDAUT    1#
ENDCLN    1#
ENDCOM    1#
ENDDU     1#
ENDHRD    1#
ENDHW     1#
ENDINI    1#
ENDMOD    1#
ENDMSG    1#
ENDPRO    1#
ENDPTA    1#
ENDRPT    1#
ENDSEG    1#
ENDSET    1#
ENDSFT    1#
ENDSRV    1#
ENDSUB    1#
```

```
ENDSW      1#
ENDTST     1#
EQUALS     1#
ERRDF      1#
ERRHRD     1#
ERROR      1#
ERRSF      1#
ERRSOF     1#
ERRTBL     1#
ESCAPE     1#
EXIT       1#
FEQUAL     1#
GETBYT     1#
GETPRI     1#
GETWOR     1#
GMANIA     1#
GMANID     1#
GMANIL     1#
GPHARD     1#
GPRMA      1#
GPRMD      1#
GPRML      1#
HEADER     1#
INLOOP     1#
IOSETU     1#
IOSTAR     1#
KT11       1#
LASTAD     1#
MANUAL     1#
MEMORY     1#
MSBYTE     1#
MSCHEC     1#
MSCNTO     1#
MSCOUN     1#
MSDATA     1#
MSDECR     1#
MSDEFA     1#
MSENDE     1#
MSERRI     1#
MSESCA     1#
MSESCS     1#
MSEXCP     1#
MSEXIT     1#
MSEXSE     1#
MSEXTJ     1#
MSGEN      1#
MSGENB     1#
MSGETS     1#
MSGETT     1#
MSGNGB     1#
MSGNIN     1#
MSGNLS     1#
MSGNSU     1#
MSGNTA     1#
MSGNTE     1#
MSHAPT     1#
```

```
MSHNAP     1#
MSINCR     1#
MSIOSE     1#
MSLDRO     1#
MSMASK     1#
MSMCHI     1#
MSMCLO     1#
MSMSK1     1#
MSPOP      1#
MSPRIN     1#
MSPUSH     1#
MSPUT      1#
MSPUT1     1#
MSRADI     1#
MSRBRO     1#
MSRNRO     1#
MSSETS     1#
MSSTAR     1#
MSSVC      1#
MSTLAB     1#
MSTSTL     1#
MSWORD     1#
MSXFER     1#
OPEN       1#
POINTE     1#
PRINTB     1#
PRINTF     1#
PRINTS     1#
PRINTX     1#
READBU     1#
READEF     1#
RFLAGS     1#
SETPRI     1#
SETVEC     1#
SLASH      1#
STARS      1#
SVC        1#
XFER       1#
XFERF      1#
XFERT      1#

. ABS.  000000     000
        000000     001
MICRG   001634     002

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 MICROG.OBJ,MICROG.LST/CR/SOL/NL:TOC=SVC34R.P11,MICROG.MAC
 RUN-TIME: 2 2 .3 SECONDS
 RUN-TIME RATIO: 32/5=6.0
 CORE USED:  31K  (61 PAGES)
```

```
      1                                .TITLE NO MORE MICROCODE MODULES
      2             000000'            .CSECT  NOMORE
      3                                .MCALL SVC
      4  000000'                       SVC
      5  000000'                       LASTAD
      6  000004'                       LSLAST::
      7             000001             .END
```

```
ASSEMB= 000010     5
C$AU  = 000052     5#
C$AUTO= 000061     5#
C$BRK = 000022     5#
C$BSEG= 000004     5#
C$BSUB= 000002     5#
C$CEFG= 000045     5#
C$CLCK= 000062     5#
C$CLEA= 000012     5#
C$CLOS= 000035     5#
C$CLP1= 000006     5#
C$CVEC= 000036     5#
C$DCLN= 000044     5#
C$DODU= 000051     5#
C$DRPT= 000024     5#
C$DU  = 000053     5#
C$EDIT= 000003     5#
C$ERDF= 000055     5#
C$ERHR= 000056     5#
C$ERRO= 000060     5#
C$ERSF= 000054     5#
C$ERSO= 000057     5#
C$ESCA= 000010     5#
C$ESEG= 000005     5#
C$ESUB= 000003     5#
C$ETST= 000001     5#
C$EXIT= 000032     5#
C$GETB= 000026     5#
C$GETW= 000027     5#
C$GMAN= 000043     5#
C$GPHR= 000042     5#
C$GPLO= 000030     5#
C$GPRI= 000040     5#
C$INIT= 000011     5#
C$INLP= 000020     5#
C$MANI= 000050     5#
C$MEM = 000031     5#
C$MSG = 000023     5#
C$OPEN= 000034     5#
C$PNTB= 000014     5#
C$PNTF= 000017     5#
C$PNTS= 000016     5#
C$PNTX= 000015     5#
C$QIO = 000377     5#
C$RDBU= 000007     5#
C$REFG= 000047     5#
C$RESE= 000033     5#
C$REVI= 000003     5#
C$RFLA= 000021     5#
C$RPT = 000025     5#
C$SEFG= 000046     5#
C$SPRI= 000041     5#
C$SVEC= 000037     5#
C$TPRI= 000013     5#
DIAGMC= 000000     5
E$END = 002100     5#
```

```
E$LOAD= 000035          5#
F$AU   = 000015         5#
F$AUTO= 000020          5#
F$BGN = 000040          5#
F$CLEA= 000007          5#
F$DU   = 000016         5#
F$END = 000041          5#
F$HARD= 000004          5#
F$HW   = 000013         5#
F$INIT= 000006          5#
F$JMP = 000050          5#
F$MOD = 000000          5#
F$MSG = 000011          5#
F$PROT= 000021          5#
F$PWR = 000017          5#
F$RPT = 000012          5#
F$SEG = 000003          5#
F$SOFT= 000005          5#
F$SRV = 000010          5#
F$SUB = 000002          5#
F$SW   = 000014         5#
F$TEST= 000001          5#
G$CNTO= 000200          5#
G$DELM= 000372          5#
G$DISP= 000003          5#
G$EXCP= 000400          5#
G$HILI= 000002          5#
G$LOLI= 000001          5#
G$NO   = 000000         5#
G$OFFS= 000400          5#
G$OFSI= 000376          5#
G$PRMA= 000001          5#
G$PRMD= 000002          5#
G$PRML= 000000          5#
G$RADA= 000140          5#
G$RADB= 000000          5#
G$RADD= 000040          5#
G$RADL= 000120          5#
G$RADO= 000020          5#
G$XFER= 000004          5#
G$YES = 000010          5#
I$AU   = 000041         5#
I$AUTO= 000041          5#
I$CLN = 000041          5#
I$DU   = 000041         5#
I$INIT= 000041          5#
I$MOD = 000041          5#
I$MSG = 000041          5#
I$PROT= 000041          5#
I$PTAB= 000041          5#
I$PWR = 000041          5#
I$RPT = 000041          5#
I$SEG = 000041          5#
I$SETU= 000041          5#
I$SRV = 000041          5#
I$SUB = 000041          5#
```

```
I$TST = 000041              5#
J$JMP = 000167              5#
L$LAST  000004RG    002     6#
O$APTS= 000000              5#
O$AU  = 000000              5#
O$BGNR= 000000              5#
O$BGNS= 000000              5#
O$DU  = 000000              5#
O$ERRT= 000000              5#
O$GNSW= 000000              5#
O$POIN= 000000              5#
O$SETU= 000000              5#      6
SVCGBL= 000000              5#      6#      7
SVCINS= 177777              5#      6
SVCSUB= 177777              5#
SVCTAG= 177777              5#
SVCTST= 177777              5#
S$LSYM= 010000              5#
T$ERRN= 000000              5#
T$GMAN= 000000              5#
T$LAST= 000001              5#      6#
T$LSYM= 010000              5#
T$LTNO= 000000              7#
T$NEST= 177777              5#
T$PTNU= 000000              5#
T$SAVL= 177777              5#
T$SEGL= 177777              5#
T$SUBN= 000000              5#
T$TAGL= 177777              5#
T$TAGN= 010000              5#
T$TEST= 000000              5#      7
T$TSTM= 177777              5#
T$TSTS= 000000              5#
X$ALWA= 000000              5#
X$FALS= 000040              5#
X$OFFS= 000400              5#
X$TRUE= 000020              5#
```

```
BCOMPL      1#      5#
BERROR      1#      5#
BGNAU       1#      5#
BGNAUT      1#      5#
BGNCLN      1#      5#
BGNDU       1#      5#
BGNHRD      1#      5#
BGNHW       1#      5#
BGNINI      1#      5#
BGNMOD      1#      5#
BGNMSG      1#      5#
BGNPRO      1#      5#
BGNPTA      1#      5#
BGNRPT      1#      5#
BGNSEG      1#      5#
BGNSET      1#      5#
BGNSFT      1#      5#
BGNSRV      1#      5#
BGNSUB      1#      5#
BGNSW       1#      5#
BGNTST      1#      5#
BNCOMP      1#      5#
BNERRO      1#      5#
BREAK       1#      5#
BRESET      1#      5#
CKLOOP      1#      5#
CLOCK       1#      5#
CLOSE       1#      5#
CLRVEC      1#      5#
COMMEN      1#      5#
DELAY       1#      5#
DESCRI      1#      5#
DEVTYP      1#      5#
DISPAT      1#      5#
DISPLA      1#      5#
DOCLN       1#      5#
DODU        1#      5#
DORPT       1#      5#
ENDAU       1#      5#
ENDAUT      1#      5#
ENDCLN      1#      5#
ENDCOM      1#      5#
ENDDU       1#      5#
ENDHRD      1#      5#
ENDHW       1#      5#
ENDINI      1#      5#
ENDMOD      1#      5#
ENDMSG      1#      5#
ENDPRO      1#      5#
ENDPTA      1#      5#
ENDRPT      1#      5#
ENDSEG      1#      5#
ENDSET      1#      5#
ENDSFT      1#      5#
ENDSRV      1#      5#
ENDSUB      1#      5#
```

ENDSW	1#	5#	
ENDTST	1#	5#	
EQUALS	1#	5#	
ERRDF	1#	5#	
ERRHRD	1#	5#	
ERROR	1#	5#	
ERRSF	1#	5#	
ERRSOF	1#	5#	
ERRTBL	1#	5#	
ESCAPE	1#	5#	
EXIT	1#	5#	
FEQUAL	1#	5#	
GETBYT	1#	5#	
GETPRI	1#	5#	
GETWOR	1#	5#	
GMANIA	1#	5#	
GMANID	1#	5#	
GMANIL	1#	5#	
GPHARD	1#	5#	
GPRMA	1#	5#	
GPRMD	1#	5#	
GPRML	1#	5#	
HEADER	1#	5#	
INLOOP	1#	5#	
IOSETU	1#	5#	
IOSTAR	1#	5#	
KT11	1#	5#	
LASTAD	1#	5#	
MANUAL	1#	5#	
MEMORY	1#	5#	
MSBYTE	1#	5#	
MSCHEC	1#	5#	
MSCNTO	1#	5#	
MSCOUN	1#	5#	
MSDATA	1#	5#	
MSDECR	1#	5#	
MSDEFA	1#	5#	
MSENDE	1#	5#	
MSERRI	1#	5#	
MSESCA	1#	5#	
MSESCS	1#	5#	
MSEXCP	1#	5#	
MSEXIT	1#	5#	
MSEXSE	1#	5#	
MSEXTJ	1#	5#	
MSGEN	1#	5#	6#
MSGENB	1#	5#	
MSGETS	1#	5#	
MSGETT	1#	5#	
MSGNGB	1#	5#	6#
MSGNIN	1#	5#	6#
MSGNLS	1#	5#	
MSGNSU	1#	5#	
MSGNTA	1#	5#	
MSGNTE	1#	5#	
MSNAPT	1#	5#	

```
MSMMAP      1#        5#
MSINCR      1#        5#
MSIOSE      1#        5#
MSLDRO      1#        5#
MSMASK      1#        5#
MSMCHI      1#        5#
MSMCLO      1#        5#
MSMSK1      1#        5#
MSPOP       1#        5#
MSPRIN      1#        5#
MSPUSH      1#        5#
MSPUT       1#        5#
MSPUT1      1#        5#
MSRADI      1#        5#
MSRBRO      1#        5#
MSRNRO      1#        5#
MSSETS      1#        5#
MSSTAR      1#        5#
MSSVC       1#        5#
MSTLAB      1#        5#
MSTSTL      1#        5#
MSWORD      1#        5#        6
MSXFER      1#        5#
OPEN        1#        5#
POINTE      1#        5#
PRINTB      1#        5#
PRINTF      1#        5#
PRINTS      1#        5#
PRINTX      1#        5#
READBU      1#        5#
READEF      1#        5#
RFLAGS      1#        5#
SETPRI      1#        5#
SETVEC      1#        5#
SLASH       1#        5#
STARS       1#        5#
SVC         1#        3#        4
XFER        1#        5#
XFERF       1#        5#
XFERT       1#        5#


. ABS.  000000      000
        000000      001
NOMORE  000004      002


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 NOMORE.OBJ,NOMORE.LST/CR/SOL/NL:TOC=SVC34R.P11,NOMORE.MAC
 RUN-TIME: 2 1 .3 SECONDS
 RUN-TIME RATIO: 42/4=9.1
 CORE USED:  31K  (61 PAGES)
```

```
LNKX11   V023   7-APR-83   17:20
#CZUAAB.BIC/B:2000,CZUAAB.MAP=CZUAAB,MICROA,MICROB,MICROC,MICROD,MICROE,MICROF,MICROG,NOMORE/E

LOAD MAP

TRANSFER ADDRESS: 000001
LOW LIMIT: 002000
HIGH LIMIT: 100564
**********
MODULE  PARAME
SECTION ENTRY    ADDRESS SIZE
<. ABS.>         000000  000000
        ADR      000020
        ADRERR   000001
        BIT0     000001
        BIT00    000001
        BIT01    000002
        BIT02    000004
        BIT03    000010
        BIT04    000020
        BIT05    000040
        BIT06    000100
        BIT07    000200
        BIT08    000400
        BIT09    001000
        BIT1     000002
        BIT10    002000
        BIT11    004000
        BIT12    010000
        BIT13    020000
        BIT14    040000
        BIT15    100000
        BIT2     000004
        BIT3     000010
        BIT4     000020
        BIT5     000040
        BIT6     000100
        BIT7     000200
        BIT8     000400
        BIT9     001000
        BOE      000400
        CLEAR    000000
        CRCSIZ   000004
        DATERR   000000
        DIM      000020
        DNI      004000
        DNIB     000010
        EF.CON   000036
        EF.NEW   000035
        EF.PWR   000034
        EF.RES   000037
        EF.STA   000040
        ENP      000400
        ERRS     040000
        EVL      000004
        FATI     000400
```

FATIB	000001
GETCMD	000002
GETPCB	000001
HOE	100000
IBE	010000
ICAB	040000
IDU	000040
IE	000100
IER	020000
INITH	177777
INITL	177777
INTE	000100
INTR	000200
IOADR	020000
IOSIZ	020000
ISR	000100
IXE	004000
LASM	000001
LIM	000021
LINADR	100000
LINSIZ	077774
LOE	040000
LOT	000010
MAXBYT	002756
MINBYT	000100
NIHLT	000006
NIUNI	000007
NPRERR	100000
NXMERR	040000
OWN	100000
PARERR	010000
PCEI	040000
PCEIB	000100
PCTO	000200
PFNOP1	000000
PFNOP2	000003
PNOP	000006
PNT	001000
POLYH	166670
POLYL	101440
POLY16	120001
PRI	002000
PRILD	000001
PRI00	000000
PRI01	000040
PRI02	000100
PRI03	000140
PRI04	000200
PRI05	000240
PRI06	000300
PRI07	000340
PSTATE	000007
RACC	000013
RACPS	000017
RC	000012
RCEI	002000
RCEIB	000004
RDPA	000002
READY	000002
RESET	000000

```
        RLSA    000024
        RMAL    000006
        RMTC    000010
        ROMADR  040000
        ROMSIZ  040000
        RPA     000004
        RPS     000016
        RRF     000010
        RSET    000040
        RSIDP   000022
        RTM     000014
        RUN     000003
        RXI     020000
        RXIB    000040
        SECOND  000077
        SERI    100000
        SERIB   000200
        SET     000001
        SFT     037400
        SFTB0   000400
        SFTB1   001000
        SFTB2   002000
        SFTB3   004000
        SFTB4   010000
        SFTB5   020000
        SIZ1K   004000
        SIZ2K   010000
        SIZ4K   020000
        SIZ8K   040000
        SLFT    000003
        STP     001000
        TXI     010000
        TXIB    000020
        UAM     000200
        UNIERR  020000
        UNIHLT  000005
        WCSADR  000000
        WCSSIZ  020000
        WLSA    000025
        WMAL    000007
        WPA     000005
        WRF     000011
        WSIDP   000023
        WTM     000015
        XPWR    100000
<         >             002000   000000
<UNAREP>                002000   053566
        BITNAM  002310
        BITNUM  002306
        BITSTA  002312
        BNAMT0  002360
        BNAMT1  002420
        BNAMT2  002460
        BTMSG   054556
        BTMSG0  055035
        BTMSG1  055073
        BTMSG2  055126
        BTMSG3  055172
        BTTBL   054676
        CHKDNI  021316
```

CHKINT	022132
CHKMON	022060
CLKBR	002276
CLKCSR	002274
CLKFRE	002302
CLKSRV	024062
CLKTAB	002274
CLKVEC	002300
CLRDNI	021362
CNTTAB	002626
COLTST	005753
CPUPRI	002676
CRCDAT	005526
CRCERR	005563
CRCPAT	005611
CRC16	021030
CRC32	020710
CSRNUM	002304
DATALD	003774
DBFRAM	004531
DEFADR	054625
DEFHDR	054562
DFPTBL	002262
DMABLK	004406
DMAFRM	004336
DMATO	004270
DPA	054542
DPPAT	004625
DWFRAM	004567
ERRINT	002670
FIFTST	005073
FORM1	006141
FORM10	006715
FORM11	007001
FORM12	007030
FORM13	007123
FORM15	007174
FORM16	007250
FORM17	007326
FORM18	007411
FORM19	007467
FORM2	006176
FORM20	007544
FORM21	007604
FORM22	007656
FORM23	007736
FORM24	010006
FORM25	010040
FORM26	010113
FORM27	010166
FORM28	010236
FORM29	010243
FORM3	006234
FORM30	010266
FORM31	010316
FORM32	010344
FORM33	010441
FORM34	010477
FORM35	010562
FORM36	010662

FORM37	010720
FORM38	011031
FORM39	011122
FORM4	006303
FORM40	011163
FORM41	011246
FORM42	011314
FORM43	011323
FORM44	011331
FORM45	011410
FORM46	011436
FORM47	011465
FORM48	011514
FORM49	011611
FORM5	006352
FORM50	011665
FORM51	011754
FORM52	012006
FORM53	012044
FORM54	012116
FORM55	012144
FORM56	012215
FORM57	012270
FORM58	012340
FORM59	012416
FORM6	006415
FORM60	012511
FORM61	012576
FORM62	012661
FORM63	012734
FORM64	012773
FORM65	013044
FORM66	013072
FORM67	013157
FORM68	013243
FORM69	013340
FORM7	006473
FORM70	013456
FORM71	013520
FORM72	013562
FORM73	013634
FORM74	013677
FORM75	013754
FORM76	014022
FORM77	014056
FORM78	014132
FORM79	014172
FORM8	006555
FORM80	014241
FORM81	014320
FORM82	014363
FORM83	014453
FORM9	006637
FREMEM	002324
FRESIZ	002322
FRSTIM	002674
HAFDUP	005723
HEXDAT	054560
HEXDPA	021116
HEXN	021200

HEXL	021236
HEXTBL	054651
HEXVAL	054561
INTBIT	004156
INTVEC	004121
LASFT	004026
LNKARB	005241
LNKBYT	004727
LNKMEM	004240
LPMSG	054554
LPMSG0	054740
LPMSG1	054777
LPTBL	054672
L$ACP	002110
L$APT	002036
L$AU	024026
L$AUT	002070
L$AUTO	023652
L$CCP	002106
L$CLEA	023654
L$CO	002032
L$DEPO	002011
L$DESC	002706
L$DESP	002076
L$DEVP	002060
L$DISP	002124
L$DLY	002116
L$DTP	002040
L$DTYP	002034
L$DU	024020
L$DUT	002072
L$DVTY	002700
L$EF	002052
L$ENVI	002044
L$ETP	002102
L$EXP1	002046
L$EXP4	002064
L$EXP5	002066
L$HARD	055256
L$HIME	002120
L$HPCP	002016
L$HPTP	002022
L$HW	002262
L$ICP	002104
L$INIT	023212
L$LADP	002026
L$LOAD	002100
L$LUN	002074
L$MREV	002050
L$NAME	002000
L$PRIO	002042
L$PROT	023204
L$PRT	002112
L$REPP	002062
L$REV	002010
L$RPT	023176
L$SPC	002056
L$SPCP	002020
L$SPTP	002024
L$STA	002030

LSTEST	002114
LSTIML	002014
LSUNIT	002012
MAXCNT	005027
METER	002332
MICMOD	002320
MICRO	002326
MSG1	014716
MSG10	015360
MSG11	015402
MSG12	015466
MSG13	015512
MSG14	015560
MSG15	015606
MSG16	015636
MSG17	015730
MSG18	016022
MSG19	016052
MSG2	014764
MSG20	016146
MSG21	016236
MSG22	016326
MSG23	016442
MSG24	016464
MSG25	016614
MSG26	017002
MSG27	017046
MSG28	017114
MSG29	017162
MSG3	015030
MSG30	017264
MSG31	017332
MSG32	017476
MSG33	017574
MSG34	017622
MSG35	017672
MSG36	017770
MSG37	020012
MSG38	020034
MSG39	020056
MSG4	015076
MSG40	020100
MSG41	020176
MSG42	020250
MSG43	020420
MSG44	020442
MSG45	020604
MSG46	020666
MSG5	015124
MSG6	015174
MSG7	015262
MSG8	015310
MSG9	015332
MUCAST	005470
NEXMEM	002666
NOCLK	023610
NOPERR	003706
ODDBYT	004765
PATERN	002520
PAT1	002520

PAT2	002522
PAT3	002524
PAT4	002526
PAT5	002530
PAT6	002532
PCBB	002606
PCOMND	002316
PCSR0	002336
PCSR0C	002350
PCSR0U	002346
PCSR1	002340
PCSR1C	002352
PCSR2	002342
PCSR2C	002354
PCSR3	002344
PCSR3C	002356
PRTPAR	006061
PWHEN	002314
RACERR	003570
RACMG1	014500
RACMG2	014526
RACMG3	014570
RACMG4	014642
RACMG7	014670
RBRRUN	005650
RCVDON	004477
RETLOG	006031
REUNA	022166
RLNKAD	005114
ROMDMP	003747
RREV	054550
RRWER	003646
RSETER	003624
SLFTST	003726
SMSG00	026610
SMSG01	026632
SMSG02	026652
SMSG03	026656
SMSG04	026706
SMSG05	026742
SMSG06	026766
SMSG07	027030
SMSG10	027073
SMSG11	027140
SMSG12	027160
SMSG13	027166
SMSG14	027213
SMSG15	027214
SMSG16	027215
SMSG17	027216
SMSG20	027217
SMSG21	027233
SMSG22	027234
SMSG23	027235
SMSG24	027236
SMSG25	027237
SMSG26	027240
SMSG27	027241
SMSG30	027242
SMSG31	027304

SMSG32	027347
SMSG33	027413
SMSG34	027450
SMSG35	027512
SMSG36	027546
SMSG37	027610
SMSG40	027644
SMSG41	027721
SMSG42	027773
SMSG43	030046
SMSG44	030112
SMSG45	030163
SMSG46	030226
SMSG47	030227
SMSG50	030230
SMSG51	030307
SMSG52	030363
SMSG53	030440
SMSG54	030506
SMSG55	030561
SMSG56	030626
SMSG57	030627
SMSG60	030630
SMSG61	030644
SMSG62	030670
SMSG63	030714
SMSG64	030734
SMSG65	030740
SMSG66	030752
SMSG67	030764
SMSG70	031000
SMSG71	031012
SMSG72	031036
SMSG73	031060
SMSG74	031061
SMSG75	031062
SMSG76	031063
SMSG77	031064
SPAT1	002534
SPAT2	002542
SPAT3	002550
SPAT4	002556
SPAT5	002564
SPAT6	002572
SPAT7	002600
STAMUX	004663
STAPAT	005305
STAPOS	005417
STAREJ	005351
STMSG	026406
STTBL	026410
SWADDR	002334
SWHDR	054706
SWPACK	054552
TDRCNT	006001
TIMOFF	021302
TIMON	021264
TIMTST	004216
TLNKAD	005165
TRAP4	024034

TRNDON	004445
T1	024076
T10	031124
T11	031540
T12	032326
T13	033006
T14	033424
T15	033714
T16	034422
T17	034740
T18	035406
T19	035764
T2	024310
T20	036344
T21	036672
T22	037202
T23	037512
T24	040072
T25	040452
T26	041046
T27	041610
T28	042206
T29	042610
T3	024642
T30	043130
T31	043520
T32	044104
T33	044470
T34	045130
T35	045460
T36	046000
T37	046350
T38	046670
T39	047230
T4	024754
T40	047620
T41	050202
T42	050634
T43	051240
T44	052330
T45	053052
T46	053466
T5	025072
T6	025560
T7	025642
T8	025740
T9	026162
UDBB	002616
UNACSR	002266
UNAINT	002672
UNAPRI	002272
UNASRV	024044
UNAVEC	002270
UNIT	002330
UNLOD	003517
UNUSED	031112
WCSMEM	004072
$ADRER	003461
$AFTER	003342
$BEFOR	003333

$BIT0	003271	
$BIT1	003263	
$BIT10	003174	
$BIT11	003165	
$BIT12	003156	
$BIT13	003147	
$BIT14	003140	
$BIT15	003131	
$BIT2	003255	
$BIT3	003247	
$BIT4	003241	
$BIT5	003233	
$BIT6	003225	
$BIT7	003217	
$BIT8	003211	
$BIT9	003203	
$CLR	003325	
$DATER	003446	
$DNI	003000	
$FATI	003021	
$GTCMD	003357	
$GTPCB	003350	
$ICAB	003076	
$INTE	003043	
$INTR	003032	
$NCLR	003313	
$NOP	003405	
$NSET	003277	
$PARER	003502	
$PATCH	055366	
$PCEI	002747	
$PCTO	003107	
$PDNDM	003417	
$RCEI	003010	
$RESET	003440	
$RMTC	003120	
$RSET	003054	
$RXI	002760	
$SERI	002736	
$SET	003307	
$SLFT	003373	
$STOP	003433	
$STRT	003411	
$TXI	002770	
$XPWR	003065	
<MICRA >	055566	000024
MICASZ	055610	
MICROA	055566	
<MICRB >	055612	002314
MICBSZ	060124	
MICROB	055612	
<MICRC >	060126	003254
MICCSZ	063400	
MICC7	063214	
MICROC	060126	
<MICRD >	063402	004706
MICDSZ	070306	
MICROD	063402	
<MICRE >	070310	002734
BFILE	072640	

```
        MICESZ  073242
        MICROE  070310
<MICRF >        073244  003460
        MICFSZ  076722
        MICROF  073244
<MICRG >        076724  001634
        MICGSZ  100556
        MICROG  076724
<NOMORE>        100560  000004
        LSLAST  100564
***********
MODULE  MICROA
SECTION ENTRY   ADDRESS SIZE
<      >        100564  000000
***********
MODULE  MICROB
SECTION ENTRY   ADDRESS SIZE
<      >        100564  000000
***********
MODULE  MICROC
SECTION ENTRY   ADDRESS SIZE
<      >        100564  000000
***********
MODULE  MICROD
SECTION ENTRY   ADDRESS SIZE
<      >        100564  000000
***********
MODULE  MICROE
SECTION ENTRY   ADDRESS SIZE
<      >        100564  000000
***********
MODULE  MICROF
SECTION ENTRY   ADDRESS SIZE
<      >        100564  000000
***********
MODULE  MICROG
SECTION ENTRY   ADDRESS SIZE
<      >        100564  000000
***********
MODULE  NO
SECTION ENTRY   ADDRESS SIZE
<      >        100564  000000


 RUN-TIME:  5 SECONDS
 5K CORE USED
```