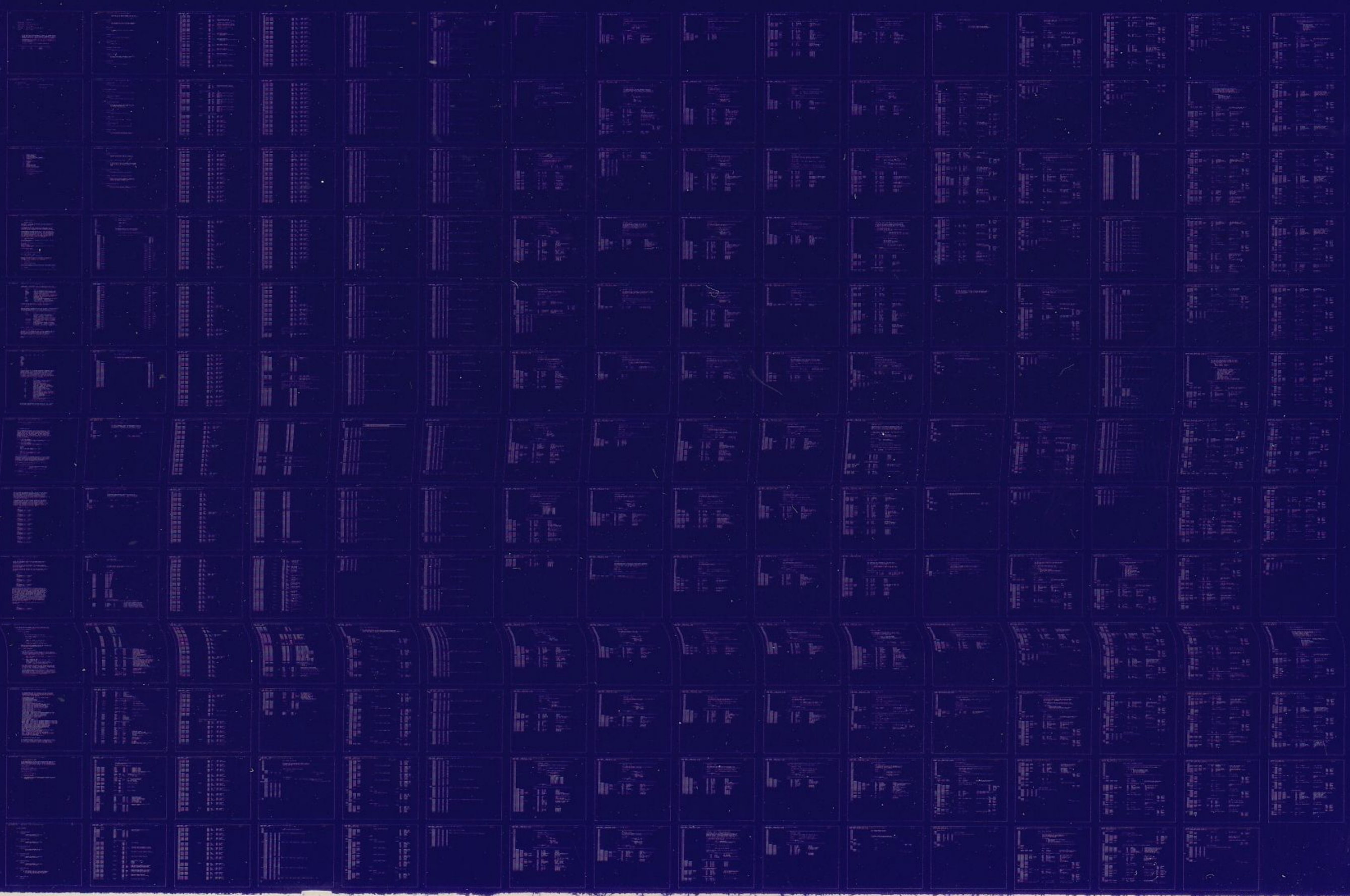


M7521  
DELUA

DELUA FUNCT DIAG  
CZUADA0

AH-T725A-MC  
1 OF 2 OCT 1985  
COPYRIGHT © 1985

digital  
MADE IN USA



The image displays a functional block diagram for the DELUA system, organized into a 10x10 grid of 100 individual blocks. Each block contains a detailed schematic of a specific functional component, such as a logic gate, a register, or a control unit. The diagrams are labeled with alphanumeric identifiers, and some include numerical values or parameters. The overall layout is dense and technical, typical of a functional block diagram for a complex digital system.

.RFM 8

IDENTIFICATION  
-----

PRODUCT CODE: AC T724A-MC  
PRODUCT NAME: CZUADAO DELUA FUNCT DIAG  
PRODUCT DATE: JULY 08, 1985  
MAINTAINER: JAMES CRITSER NAC SOFTWARE ENG MKO  
AUTHOR: JOHN C. CARMODY

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C) 1985 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	

REVISION HISTORY

DATE -----	AUTHOR -----	REASON/DESCRIPTION OF CHANGE -----
08 JUL 85	J. CARMODY	INITIAL RELEASE

## TABLE OF CONTENTS

1.0	GENERAL INFORMATION
1.1	PROGRAM ABSTRACT
1.2	SYSTEM REQUIREMENTS
1.3	RELATED DOCUMENTS AND STANDARDS
1.4	DIAGNOSTIC HIERARCHY PREREQUISITES
1.5	ASSUMPTIONS
2.0	OPERATING INSTRUCTIONS
2.1	COMMANDS
2.2	SWITCHES
2.3	FLAGS
2.4	HARDWARE QUESTIONS
2.5	SOFTWARE QUESTIONS
2.6	EXTENDED P-TABLE DIALOGUE
2.7	QUICK STARTUP PROCEDURE
3.0	ERROR INFORMATION
4.0	PERFORMANCE AND PROGRESS REPORTS
5.0	DEVICE INFORMATION TABLES
6.0	TEST SUMMARIES

## 1.0 GENERAL INFORMATION

### 1.1 PROGRAM ABSTRACT

THIS PRODUCT IS THE PDP-11 FUNCTIONAL TESTING DIAGNOSTIC FOR THE DELUA. A CONFIGURATION OF UP TO EIGHT DELUA UNITS WILL BE ACCEPTED FOR TEST.

THIS DIAGNOSTIC WILL ONLY OPERATE IN A STAND ALONE, OFFLINE ENVIRONMENT USING THE DELUA OPERATIONAL MICROCODE. FAILURE IDENTIFICATION WILL GENERALLY BE TO THE FAILING DELUA FUNCTION.

THIS DIAGNOSTIC HAS BEEN WRITTEN FOR USE WITH THE DIAGNOSTIC RUNTIME SERVICES SOFTWARE (SUPERVISOR). THESE SERVICES PROVIDE THE INTERFACE TO THE OPERATOR AND TO THE SOFTWARE ENVIRONMENT. THIS PROGRAM CAN BE USED WITH XXDP+, ACT, APT, 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 HARDWARE IS REQUIRED TO RUN THE DELUA FUNCTIONAL TESTING DIAGNOSTIC:

PDP-11 CPU FROM SUPPORTED LIST (SEE BELOW)  
16K MEMORY  
CONSOLE TERMINAL  
DELUA, WITH H4080 LOOPBACK CONNECTOR INSTALLED, IF PLAN TO RUN EXTERNAL LOOPBACK TEST.

SUPPORTED PDP-11 CPU'S:  
11/24, 11/34A, 11/44, 11/70, 11/84

### 1.3 RELATED DOCUMENTS AND STANDARDS

XXDP+ USER'S MANUAL - CHQUS

### 1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

THE TESTS INCLUDED IN THIS DELUA FUNCTIONAL DIAGNOSTIC ARE ARRANGED IN A TEST HIERARCHY. TESTS SHOULD BE EXECUTED IN CONSECUTIVE ORDER FOR MAXIMUM FAULT ISOLATION.

### 1.5 ASSUMPTIONS

## 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 TEST 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
PRINT	PRINT STATISTICAL INFORMATION (IF IMPLEMENTED BY THE DIAGNOSTIC - SECTION 4.0)
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 "DDDD".

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:DDDD	EXECUTE DDDDD PASSES (DDDD = 1 TO 64000)
/FLAGS:FLGS	SET SPECIFIED FLAGS. FLAGS ARE DESCRIBED IN SECTION 2.3.
/EOP:DDDD	REPORT END OF PASS MESSAGE AFTER EVERY DDDDD PASSES ONLY. (DDDD = 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
PRINT					
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)
ISR	INHIBIT STATISTICAL REPORTS (DOES NOT APPLY TO DIAGNOSTICS WHICH DO NOT SUPPORT STATISTICAL REPORTING)
IDR	INHIBIT PROGRAM DROPPING OF UNITS
ADR	EXECUTE AUTODROP CODE
LOT	LOOP ON TEST
EVL	EXECUTE EVALUATION (ON DIAGNOSTICS WHICH HAVE EVALUATION SUPPORT)

\*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.  
THE ALLOWABLE RANGE IS 174510 - 174600 OCTAL.

WHAT IS THE VECTOR ADDRESS ?

THIS IS THE INTERRUPT VECTOR ADDRESS FOR THIS DEVICE.  
THE ALLOWABLE RANGE IS 120 - 210 OCTAL.

SAMPLE DIALOGUE:

UNIT 0

WHAT IS THE PCSRO ADDRESS? (0) ? 174510

WHAT IS THE VECTOR ADDRESS? (0) ? 120

UNIT 1

WHAT IS THE PCSRO ADDRESS? (0) ? 174520

WHAT IS THE VECTOR ADDRESS (0) ? 130

#### 2.5 SOFTWARE QUESTIONS

AFTER YOU HAVE ANSWERED THE HARDWARE QUESTIONS OR AFTER A RESTART OR CONTINUE COMMAND, THE RUNTIME SERVICES WILL ASK FOR SOFTWARE PARAMETERS. THESE PARAMETERS WILL GOVERN SOME DIAGNOSTIC SPECIFIC OPERATION MODES. YOU WILL BE PROMPTED BY "CHANGE SW (L) ?" IF YOU WISH TO CHANGE ANY PARAMETERS, ANSWER BY TYPING "Y".

THE FOLLOWING IS THE ONLY SOFTWARE QUESTION FOR THIS DEVICE:

RUN TEST 26 EXTERNAL LOOPBACK TEST?

THE DEFAULT IS NO ( No skips test 26, This means that  
External Loopback will not be tested  
at all).

YES, WILL LOOP A FRAME USING EXTERNAL LOOPBACK MODE.

SAMPLE DIALOGUE:

RUN TEST 26 EXTERNAL LOOPBACK TEST ? (L) N ? Y <CR>

#### 2.6 EXTENDED P-TABLE DIALOGUE

WHEN YOU ANSWER THE HARDWARE QUESTIONS, YOU ARE BUILDING ENTRIES IN A TABLE THAT DESCRIBES THE DEVICES UNDER TEST. THE SIMPLEST WAY TO BUILD THIS TABLE IS TO ANSWER ALL QUESTIONS FOR EACH UNIT TO BE TESTED. IF YOU HAVE A MULTIPLEXED DEVICE SUCH AS A MASS STORAGE CONTROLLER WITH SEVERAL DRIVES OR A COMMUNICATION DEVICE WITH SEVERAL LINES, THIS BECOMES TEDIOUS SINCE MOST OF THE ANSWERS ARE REPETITIOUS.

TO ILLUSTRATE A MORE EFFICIENT METHOD, SUPPOSE YOU ARE TESTING A FICTIONAL DEVICE, THE XY11. SUPPOSE THIS DEVICE CONSISTS OF A CONTROL MODULE WITH EIGHT UNITS (SUB-DEVICES) ATTACHED TO IT. THESE UNITS ARE DESCRIBED BY THE OCTAL NUMBERS 0 THROUGH 7. THERE IS ONE HARDWARE PARAMETER THAT CAN VARY AMONG UNITS CALLED THE Q-FACTOR. THIS Q-FACTOR MAY BE 0 OR 1. BELOW IS A SIMPLE WAY TO BUILD A TABLE FOR ONE XY11 WITH EIGHT UNITS.

\* UNITS (0) ? 8<CR>

UNIT 1  
CSR ADDRESS (0) ? 174510<CR>  
SUB-DEVICE # (0) ? 0<CR>  
Q-FACTOR (0) 0 ? 1<CR>

UNIT 2  
CSR ADDRESS (0) ? 174510<CR>  
SUB-DEVICE # (0) ? 1<CR>  
Q-FACTOR (0) 1 ? 0<CR>

UNIT 3  
CSR ADDRESS (0) ? 174510<CR>  
SUB-DEVICE # (0) ? 2<CR>  
Q-FACTOR (0) 0 ? <CR>

UNIT 4  
CSR ADDRESS (0) ? 174510<CR>  
SUB-DEVICE # (0) ? 3<CR>  
Q-FACTOR (0) 0 ? <CR>

UNIT 5  
CSR ADDRESS (0) ? 174510<CR>  
SUB-DEVICE # (0) ? 4<CR>  
Q-FACTOR (0) 0 ? <CR>

UNIT 6  
CSR ADDRESS (0) ? 174510<CR>  
SUB-DEVICE # (0) ? 5<CR>  
Q-FACTOR (0) 0 ? <CR>

UNIT 7  
CSR ADDRESS (0) ? 174510<CR>  
SUB-DEVICE # (0) ? 6<CR>  
Q-FACTOR (0) 0 ? 1<CR>

UNIT 8  
CSR ADDRESS (0) 174510<CR>  
SUB-DEVICE # (0) ? 7<CR>

Q-FACTOR (0) 1 ? <CR>

NOTICE THAT THE DEFAULT VALUE FOR THE Q-FACTOR CHANGES WHEN A NON-DEFAULT RESPONSE IS GIVEN. BE CAREFUL WHEN SPECIFYING MULTIPLE UNITS!

AS YOU CAN SEE FROM THE ABOVE EXAMPLE, THE HARDWARE PARAMETERS DO NOT VARY SIGNIFICANTLY FROM UNIT TO UNIT. THE PROCEDURE SHOWN IS NOT VERY EFFICIENT.

THE RUNTIME SERVICES CAN TAKE MULTIPLE UNIT SPECIFICATIONS HOWEVER. LET'S BUILD THE SAME TABLE USING THE MULTIPLE SPECIFICATION FEATURE.

♦ UNITS (0) ? 8<CR>

UNIT 1  
CSR ADDRESS (0) ? 174510<CR>  
SUB-DEVICE ♦ (0) ? 0,1<CR>  
Q-FACTOR (0) 0 ? 1,0<CR>

UNIT 3  
CSR ADDRESS (0) ? 174510<CR>  
SUB-DEVICE ♦ (0) ? 2-5<CR>  
Q-FACTOR (0) 0 ? 0<CR>

UNIT 7  
CSR ADDRESS (0) ? 174510<CR>  
SUB-DEVICE ♦ (0) ? 6,7<CR>  
Q-FACTOR (0) 0 ? 1<CR>

AS YOU CAN SEE IN THE ABOVE DIALOGUE, THE RUNTIME SERVICES WILL BUILD AS MANY ENTRIES AS IT CAN WITH THE INFORMATION GIVEN IN ANY ONE PASS THROUGH THE QUESTIONS. IN THE FIRST PASS, TWO ENTRIES ARE BUILT SINCE TWO SUB-DEVICES AND Q-FACTORS WERE SPECIFIED. THE SERVICES ASSUME THAT THE CSR ADDRESS IS 174510 FOR BOTH SINCE IT WAS SPECIFIED ONLY ONCE. IN THE SECOND PASS, FOUR ENTRIES WERE BUILT. THIS IS BECAUSE FOUR SUB-DEVICES WERE SPECIFIED. THE "-" CONSTRUCT TELLS THE RUNTIME SERVICES TO INCREMENT THE DATA FROM THE FIRST NUMBER TO THE SECOND. IN THIS CASE, SUB-DEVICES 2, 3, 4 AND 5 WERE SPECIFIED. (IF THE SUB-DEVICE WERE SPECIFIED BY ADDRESSES, THE INCREMENT WOULD BE BY 2 SINCE ADDRESSES MUST BE ON AN EVEN BOUNDARY.) THE CSR ADDRESSES AND Q-FACTORS FOR THE FOUR ENTRIES ARE ASSUMED TO BE 174510 AND 0 RESPECTIVELY SINCE THEY WERE ONLY SPECIFIED ONCE. THE LAST TWO UNITS ARE SPECIFIED IN THE THIRD PASS.

THE WHOLE PROCESS COULD HAVE BEEN ACCOMPLISHED IN ONE PASS AS SHOWN BELOW.

♦ UNITS (0) ? 8<CR>

UNIT 1  
CSR ADDRESS (0) ? 174510<CR>  
SUB-DEVICE ♦ (0) ? 0-7<CR>  
Q-FACTOR (0) 0 ? 0,1,0,...,1,1<CR>

AS YOU CAN SEE FROM THIS EXAMPLE, NULL REPLIES (COMMAS ENCLOSING A NULL FIELD) TELL THE RUNTIME SERVICES TO REPEAT THE LAST REPLY.

## 2.7 QUICK START-UP PROCEDURE (XXDP+)

TO START-UP THIS PROGRAM:

1. BOOT XXDP+
2. GIVE THE DATE AND ANSWER ANY QUESTIONS
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
7. ANSWER THE "CHANGE SW" QUESTION WITH "N"

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 "IBE" 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.

### 3.2 SPECIFIC ERROR MESSAGES

ALL ERROR REPORTS FOR THIS DIAGNOSTIC ARE SELF-EXPLANATORY AND WHENEVER POSSIBLE CALLS OUT THE FAILING DELUA FUNCTION. WHENEVER A DATA COMPARE ERROR IS REPORTED THE "SHOULD BE" AND "WAS" DATA WILL ALSO BE REPORTED.

THE FOLLOWING IS A LIST OF ALL THE POSSIBLE ERRORS:

REGISTER ACCESS ERROR  
DATA COMPARE ERROR IN PCSR2  
DATA COMPARE ERROR IN PCSR3  
DNI BIT FAILED TO SET AFTER DEVICE RESET  
SELF TEST FAILURE  
WRITING ONE TO CLEAR DNI BIT FAILED  
NO DNI INTERRUPT OCCURED AFTER GET PCBB PORT COMMAND  
DNI BIT FAILED TO SET AFTER NOP PORT COMMAND  
DNI BIT FAILED TO SET AFTER GET PCBB PORT COMMAND  
DNI BIT FAILED TO SET AFTER GET CMD PORT COMMAND  
DNI BIT FAILED TO SET AFTER START PORT COMMAND  
TXI BIT FAILED TO SET  
WRITING ONE TO CLEAR TXI BIT FAILED  
RXI BIT FAILED TO SET  
WRITING ONE TO CLEAR RXI BIT FAILED  
TIMEOUT ERROR - DELUA FAILED TO RELINQUISH OWNERSHIP OF RDRB  
TIMEOUT ERROR - DELUA FAILED TO RELINQUISH OWNERSHIP OF TORB  
DNI BIT FAILED TO SET AFTER STOP PORT COMMAND  
DATA COMPARE ERROR IN TRANSMIT DESCRIPTOR RING  
DATA COMPARE ERROR IN RECEIVE DESCRIPTOR RING  
TRANSMIT-RECEIVE DATA COMPARE ERROR  
CRC COMPARE ERROR  
INTERNAL ROM CRC COMPARE ERROR  
RCBI BIT FAILED TO SET  
TIMEOUT ERROR - DELUA FAILED TO RELINQUISH OWNERSHIP OF FIRST TORB  
TIMEOUT ERROR - DELUA FAILED TO RELINQUISH OWNERSHIP OF SECOND TORB  
TIMEOUT ERROR - DELUA FAILED TO RELINQUISH OWNERSHIP OF FIRST RDRB  
TIMEOUT ERROR - DELUA FAILED TO RELINQUISH OWNERSHIP OF SECOND RDRB  
DATA COMPARE ERROR IN FIRST TRANSMIT DESCRIPTOR RING  
DATA COMPARE ERROR IN SECOND TRANSMIT DESCRIPTOR RING  
DATA COMPARE ERROR IN FIRST RECEIVE DESCRIPTOR RING  
DATA COMPARE ERROR IN SECOND RECEIVE DESCRIPTOR RING  
DNI BIT NOT SET AFTER PORT HALT COMMAND  
FATAL ERROR - DELUA ID BIT NOT SET  
ERROR - LOOPBACK SUCCESSFUL WITH INVALID DESTINATION ADDRESS  
INTERNAL RAM MEMORY DATA COMPARE ERROR  
DNI BIT FAILED TO SET AFTER SELF TEST PORT COMMAND  
'BUFL', IN TORB+6 NOT SET ON XMIT BUFF OVERFLOW WITH <DTCR=0>  
'BUFL', IN TORB+6 NOT SET ON XMIT BUFF OVERFLOW WITH <DTCR=1>  
PCSR0 INTERRUPT BIT CLEAR ERROR  
RECEIVED PACKET COUNTER NOT GREATER THAN 0

#### 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 FOR EACH DEVICE BEING TESTED DEVICE INFORMATION FOR THAT DEVICE IS PRINTED. THIS PRINTOUT CONTAINS THE ETHERNET DEFAULT ADDRESS, THE ROM MICROCODE VERSION, AND THE SWITCH PACK SETTINGS FOR SELF TEST LOOP AND REMOTE BOOT.

## 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

NOTE: THIS INFORMATION MAY BE PRINTED WITHOUT RUNNING THE ENTIRE DIAGNOSTIC IF TEST 27 IS RUN SEPARATELY VIA THE /TESTS:27 SUPERVISOR SWITCH.

## 6.0 TEST SUMMARIES

## TEST 1: PCSRO READ ACCESS

VERIFIES:  
A DEVICE IS PRESENT AT THE PCSRO  
UNIBUS ADDRESS SPECIFIED.

## TEST 2: PCSR1 READ ACCESS

VERIFIES:  
A DEVICE IS PRESENT AT THE PCSR1  
UNIBUS ADDRESS SPECIFIED.

## TEST 3: PCSR1 DELUA ID BIT

VERIFIES:  
BIT 06, AND NO OTHER BITS IN THE  
PCSR1 DEVICE ID FIELD IS SET.

## TEST 4: PCSR2 READ ACCESS

VERIFIES:  
A DEVICE IS PRESENT AT THE PCSR2  
UNIBUS ADDRESS SPECIFIED.

## TEST 5: PCSR3 READ ACCESS

VERIFIES:  
A DEVICE IS PRESENT AT THE PCSR3  
UNIBUS ADDRESS SPECIFIED.

## TEST 6: PCSR2 STATIC BIT

VERIFIES:  
PCSR2 FOR ALL STUCK-AT-0 (SA0) AND STUCK-AT-1  
(SA1) ERRORS. THE HOST WILL WRITE PATTERNS  
TO PCSR2, AND READ THEM BACK TO VERIFY.

## TEST 7: PCSR3 STATIC BIT

VERIFIES:

PCSR3 FOR ALL SA0 AND SA1 ERRORS. THE HOST WILL  
WRITE PATTERNS TO PCSR3 AND READ THEM BACK TO VERIFY.

TEST 8: SELF TEST

VERIFIES:

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

TEST 9: PORT COMMAND

VERIFIES:

NO ERRORS OCCUR WHEN A DELUA PORT COMMAND IS  
ISSUED.

TEST 10: INTERRUPT LOGIC

VERIFIES:

A DELUA INTERRUPT CAN BE GENERATED.

TEST 11: READ INTERNAL ROM

VERIFIES:

INTERNAL ROM.

TEST 12: READ/WRITE INTERNAL MEMORY

VERIFIES:

INTERNAL RAM CAN BE WRITTEN AND READ

TEST 13: INTERNAL LOOPBACK

VERIFIES:

NO ERRORS OCCUR WHEN A DATAGRAM IS TRANSMITTED  
AND RECEIVED IN INTERNAL LOOPBACK MODE.

TEST 14: CRC CHECKING

VERIFIES:

CRC CHECKING LOGIC IS OPERATIONAL.



TEST 15: FORCE CRC ERROR

VERIFIES:  
CRC ERROR DETECTION IS OPERATIONAL.

TEST 16: NO RECEIVE BUFFER

VERIFIES:  
A RECEIVE BUFFER ERROR (RCBI) CAN BE GENERATED.

TEST 17: DISABLE RECEIVE CHAINING

VERIFIES:  
DISABLE RECEIVE CHAINING MODE IS OPERATIONAL.

TEST 18: TRANSMIT CHAINING ERROR

VERIFIES:  
DETECTION OF A BUFFER LENGTH ERROR WILL CAUSE  
THE CORRESPONDING ERROR BIT, 'BUFL' TO SET IN  
THE TRANSMIT DESCRIPTOR RING.

TEST 19: DATA CHAINING

VERIFIES:  
TRANSMIT AND RECEIVE DATA CHAINING.

TEST 20: PHYSICAL ADDRESS

VERIFIES:  
PHYSICAL ADDRESS FUNCTION IS OPERATIONAL.

TEST 21: MULTICAST ADDRESS

VERIFIES:  
MULTICAST ADDRESS FUNCTION IS OPERATIONAL.

TEST 22: PROMISCUOUS ADDRESS

VERIFIES:  
THE DELUA IN PROMISCUOUS MODE WILL ACCEPT ALL  
PACKETS REGARDLESS OF DESTINATION ADDRESS.

## TEST 23: ENABLE ALL MULTICAST

## VERIFIES:

THE DELUA IN MULTICAST MODE WILL ACCEPT ALL PACKETS WITH MULTICAST DESTINATION ADDRESSES.

## TEST 24: INT. LOOPBACK TRANSMIT LENGTH ERROR

## VERIFIES:

IF PORT DRIVER ATTEMPTS TO TRANSMIT GREATER THAN A 32 BYTE <DTCR = 0> OR 36 BYTE <DTCR = 1> TRANSMIT FRAME, THE DEVICE WILL RETURN A 'TRANSMIT LENGTH' ERROR.

## TEST 25: SIMULTANEOUS OPERATIONS

## VERIFIES:

SIMULTANEOUS OPERATIONS CAN BE PERFORMED.

## TEST 26: EXTERNAL LOOPBACK (MANUAL INTERVENTION REQUIRED)

## VERIFIES:

USING AN H4080 LOOPBACK CONNECTOR, INSURES NO ERRORS OCCUR WHEN A DATAGRAM IS TRANSMITTED AND RECEIVED IN EXTERNAL LOOPBACK MODE.

## TEST 27: PRINT DEVICE PARAMETERS

## VERIFIES:

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

```

810 .TITLE PROGRAM HEADER AND TABLES
811
812 .SBTTL PROGRAM HEADER
830
832 ; .ENABL ABS,AMA
833 ; .ENABL AMA
834 ; . = 2000
836
837 000000 BGNMOD
838
839 ;++
840 ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
841 ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
842 ;--
843
844 000000 POINTER BGNRPT,BGNSW,BGNSFT,BGNAU,BGNDU,ERRTBL
845
846
847 000000 HEADER CZUADA,A,0,11,0,340

```

000000	
000000	103
000001	132
000002	125
000003	101
000004	104
000005	101
000006	000
000007	000
000010	
000010	101
000011	
000011	060
000012	
000012	000000
000014	
000014	000011
000016	
000016	102676'
000020	
000020	103010'
000022	
000022	000214'
000024	
000024	000222'
000026	
000026	103132'
000030	
000030	000000
000032	
000032	000000
000034	
000034	000000
000036	
000036	000000
000040	
000040	000124'
000042	

```

L$NAME::
        .ASCII /C/
        .ASCII /Z/
        .ASCII /U/
        .ASCII /A/
        .ASCII /D/
        .ASCII /A/
        .BYTE 0
        .BYTE 0
L$REV::
        .ASCII /A/
L$DEPO::
        .ASCII /O/
L$UNIT::
        .WORD 0
L$TIML::
        .WORD 11
L$MPCP::
        .WORD L$HARD
L$SPCP::
        .WORD L$SOFT
L$HPTP::
        .WORD L$HW
L$SPTP::
        .WORD L$SW
L$LADP::
        .WORD L$LAST
L$STA::
        .WORD 0
L$CO::
        .WORD 0
L$DTYP::
        .WORD 0
L$APT::
        .WORD 0
L$DTP::
        .WORD L$DISPATCH
L$PRIO::

```

000042 000340  
000044  
000044 000000  
000046  
000046 000000  
000050  
000050 004  
000051 000  
000052  
000052 000000  
000054 000000  
000056  
000056 000000  
000060  
000060 016660'  
000062  
000062 033514'  
000064  
000064 000000  
000066  
000066 000000  
000070  
000070 034356'  
000072  
000072 034350'  
000074  
000074 000000  
000076  
000076 016666'  
000100  
000100 104035  
000102  
000102 016650'  
000104  
000104 033530'  
000106  
000106 034342'  
000110  
000110 034340'  
000112  
000112 033522'  
000114  
000114 000000  
000116  
000116 000000  
000120  
000120 000000

.WORD 340  
L\$ENVI::  
.WORD 0  
L\$EXP1::  
.WORD 0  
L\$MREV::  
.BYTE C\$REVISION  
.BYTE C\$EDIT  
L\$EF::  
.WORD 0  
.WORD 0  
L\$SPC::  
.WORD 0  
L\$DEVP::  
.WORD L\$DVTYP  
L\$REPP::  
.WORD L\$RPT  
L\$EXP4::  
.WORD 0  
L\$EXP5::  
.WORD 0  
L\$AUT::  
.WORD L\$AU  
L\$DUT::  
.WORD L\$DU  
L\$LUN::  
.WORD 0  
L\$DESP::  
.WORD L\$DESC  
L\$LOAD::  
EMT E\$LOAD  
L\$ETP::  
.WORD L\$ERRTBL  
L\$ICP::  
.WORD L\$INIT  
L\$CCP::  
.WORD L\$CLEAN  
L\$ACP::  
.WORD L\$AUTO  
L\$PRT::  
.WORD L\$PROT  
L\$TEST::  
.WORD 0  
L\$DLY::  
.WORD 0  
L\$HIME::  
.WORD 0

848  
849

851  
852  
853  
854  
855  
856  
857  
858 000122  
000122 000033  
000124  
000124 034444'  
000126 034624'  
000130 035012'  
000132 035220'  
000134 035406'  
000136 035574'  
000140 036012'  
000142 036230'  
000144 040166'  
000146 040616'  
000150 041170'  
000152 041720'  
000154 043160'  
000156 044574'  
000160 046156'  
000162 047576'  
000164 050704'  
000166 052322'  
000170 053614'  
000172 055424'  
000174 061106'  
000176 065042'  
000200 071302'  
000202 073600'  
000204 075244'  
000206 077134'  
000210 101000'

.SBTTL DISPATCH TABLE

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

DISPATCH 27.

.WORD 27  
L\$DISPATCH::  
.WORD T1  
.WORD T2  
.WORD T3  
.WORD T4  
.WORD T5  
.WORD T6  
.WORD T7  
.WORD T8  
.WORD T9  
.WORD T10  
.WORD T11  
.WORD T12  
.WORD T13  
.WORD T14  
.WORD T15  
.WORD T16  
.WORD T17  
.WORD T18  
.WORD T19  
.WORD T20  
.WORD T21  
.WORD T22  
.WORD T23  
.WORD T24  
.WORD T25  
.WORD T26  
.WORD T27

859

861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874

000212  
000212 000002  
000214  
000214  
000214  
000214  
000214 000000  
000216 000000  
000220  
000220

.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.  
;-

BGNHW DFPTBL

.WORD L10000-L\$HW/2

L\$HW::  
DFPTBL::

.WORD 0  
.WORD 0  
ENDHW

; PCSRO - UNIBUS ADDRESS  
; DELUA INTERRUPT VECTOR

L10000:

```
876      .SBTTL  SOFTWARE P-TABLE
877
878      ;**
879      ; THE SOFTWARE TABLE CONTAINS VARIOUS DATA USED BY THE
880      ; PROGRAM AS OPERATIONAL PARAMETERS.  THESE PARAMETERS ARE
881      ; SET UP AT ASSEMBLY TIME AND MAY BE VARIED BY THE OPERATOR
882      ; AT RUN TIME.
883      ;-
884
885      BGNSW  SFPTBL
886
887      000220 000001
888      000222
889      000222
890      000222
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
```

.WORD L10001-L10001/2  
L10001:  
SFPTBL::  
EXLOOP: .WORD 0 ; EXTERNAL LOOPBACK FLAG  
.EVEN  
ENDSW  
L10001:

905  
 906  
 926  
 927  
 928  
 929  
 930  
 931  
 932  
 933  
 934  
 935  
 936 000224

.TITLE GLOBAL AREAS  
 .SBTTL GLOBAL EQUATES SECTION

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

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.	; BIT POSITION IN SECOND STATUS WORD
000037	EF.RESTART== 31.	; (100000) START COMMAND WAS ISSUED
000036	EF.CONTINUE== 30.	; (040000) RESTART COMMAND WAS ISSUED
000035	EF.NEW== 29.	; (020000) CONTINUE COMMAND WAS ISSUED
000034	EF.PWR== 28.	; (010000) A NEW PASS HAS BEEN STARTED
		; (004000) 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
;
; PCSRO - PORT CONTROL AND STATUS REGISTER 0
937          ;
938          SERI    ==    BIT15      ; STATUS ERROR INTERRUPT
939          PCEI    ==    BIT14      ; PORT COMMAND ERROR INTERRUPT
940          RXI     ==    BIT13      ; RECEIVE RING INTERRUPT
941          TXI     ==    BIT12      ; TRANSMIT RING INTERRUPT
942          DNI     ==    BIT11      ; DONE INTERRUPT
943          RCBI    ==    BIT10      ; RECEIVE BUFFER UNAVAILABLE
944          FATL    ==    BIT09      ; FATAL ERROR INTERRUPT -
945          ; TELL PORT DRIVER TO IGNORE CONTENTS
946          ; OF PCSR1
947          ; UNSOLICITED STATE CHANGE INTERRUPT
948          USCI    ==    BIT08      ; UNSOLICITED STATE CHANGE INTERRUPT
949          CLINTB  ==    SERI+PCEI+RXI+TXI+DNI+FATL+USCI
950          ; WRITE 1 TO CLEAR MASK - PCSRO UPPER BYTE
951          SERIB   ==    BIT07      ; STATUS ERROR INTERRUPT BYTE REFERENCE
952          PCEIB   ==    BIT06      ; PORT COMMAND ERROR INTERRUPT BYTE REF
953          RXIB    ==    BIT05      ; RECEIVE RING INTERRUPT BYTE REF
954          TXIB    ==    BIT04      ; TRANSMIT RING INTERRUPT BYTE REF
955          DNIB    ==    BIT03      ; DONE INTERRUPT BYTE REF
956          RCBIB   ==    BIT02      ; RECEIVE BUFFER UNAVAILABLE
957          FATLIB   ==    BIT01      ; FATAL ERROR INTERRUPT BYTE REF.
958          USCIB   ==    BIT00      ; UNSOLICITED STATE CHANGE INTERRUPT BYTE REF.
959          ;
960          INTR    ==    BIT07      ; INTERRUPT SUMMARY <15:08>
961          INTE    ==    BIT06      ; INTERRUPT ENABLE
962          RSET    ==    BIT05      ; DELUA RESET
963          ;
964          ;DEVICE ID <06:04>          ;IDENTIFIES DEVICE TO HOST
965          ; BIT04 SET ONLY = DELUA
966          ;PORT COMMANDS <03:00>
967          GETPCB  ==    BIT00
968          GETCMD  ==    BIT01
    
```

```

969      000003      SLFT      ==      BIT00!BIT01
970      000004      START     ==      BIT02
971      000006      PNOP      ==      BIT01!BIT02
972      0C0010      PDMD      ==      BIT03
973      000016      HALT     ==      BIT03!BIT02!BIT01
974      000017      STOP     ==      BIT03!BIT02!BIT01!BIT00
975      ;
976      ;PCSR1      PORT CONTROL AND STATUS REGISTER 1
977      ;
978      ;SELF TEST ERROR CODE <13:08>
979      140377      STMASK   ==      140377      ; SELF TEST MASK
980      ;
981      000200      PCTO     ==      BIT07      ; PORT COMMAND TIMEOUT
982      ;
983      000010      RMTC     ==      BIT03      ; REMOTE CONSOLE RESERVED
984      ;
985      ;
986      ;DEVICE ID FIELD <06:04>
987      000020      DELUAT  ==      BIT04      ;DEVICE IS DELUA IF ONLY BIT SET
988      ;
989      ;
990      ;PORT STATE <02:00>
991      177770      SMASK   ==      177770      ; STATE MASK
992      ;
993      000000      RESET   ==      0
994      000001      PRILD   ==      BIT00      ; PRIMARY LOAD STATE
995      000002      READY   ==      BIT01
996      000003      RUN     ==      BIT00!BIT01
997      000005      UNIHLT  ==      BIT00!BIT02
998      000006      NIHLT   ==      BIT01!BIT02
999      000007      NIUNI   ==      BIT00!BIT01!BIT02
1000     ;
1001     ;DESCRIPTOR RING DEFINITIONS
1002     100000      OWN     ==      BIT15
1003     040000      ERRS    ==      BIT14
1004     001000      STP     ==      BIT09
1005     000400      ENP     ==      BIT08
1006     ;
1007     100000      BUFL    ==      BIT15
1008     ;GLOBAL EQUATES
1009     000000      ZERO    ==      0
1010     177777      ONES    ==      177777
1011     000377      TIMASK  ==      377      ; UPPER BYTE = ONES
1012     000000      GOODST  ==      0      ; SUCCESSFUL SELF TEST CODE
1013     172377      STATEM  ==      172377 ; MASK ALL PCSR0 BITS EXCEPT STATE BITS
1014     175015      CMODE1  ==      175015 ; ALL SETABLE MODE BITS = ONES
1015     007777      TDRMSK  ==      7777      ; TDR MASK
1016     002540      DTYPE   ==      2540      ; DIAGNOSTIC TYPE FIELD
1017     ;
1018     000000      INITH   ==      0      ; INITIAL CRC VALUE
1019     ;POLYH      ==      120001 ; CRC POLYNOMIAL
1020     120001      POLYHI  ==      120001 ;CRC POLYNOMIAL
1021     ;
1022     020000      SIZ4K   ==      20000      ; 4K WORDS
1023     040000      SIZ8K   ==      SIZ4K*2 ; 8K WORDS
1024     000077      SECOND  ==      63.      ;63 LINE CLOCK TICKS = APROX. 1 SECOND
1025     000100      IE      ==      100      ;INTERRUPT ENABLE FOR LINE CLOCK
    
```

```

1027          .SBTTL GLOBAL DATA SECTION
1028
1029          ;**
1030          ; THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
1031          ; IN MORE THAN ONE TEST.
1032          ;
1033          ; ADDRESSES FOR DELUA UNDER TEST
1034          ;
1035 000224 000000 PCSRO:      .WORD 0      ; ADDRESS OF PCSRO
1036 000226 000000 PCSR1:      .WORD 0      ; ADDRESS OF PCSR1
1037 000230 000000 PCSR2:      .WORD 0      ; ADDRESS OF PCSR2
1038 000232 000000 PCSR3:      .WORD 0      ; ADDRESS OF PCSR3
1039 000234 000000 PCUPB:      .WORD 0      ; ADDRESS OF THE UPPER BYTE OF PCSRO
1040 000236 000000 PCSRO:      .WORD 0      ; PCSRO DATA SAVE LOCATION
1041          ;
1042 000240 000000 INTVEC:   .WORD 0      ; ADDRESS OF DELUA INTERRUPT VECTOR
1043 000242 000240 UNAPRI:   .WORD 240   ; UNA PRIORITY = 5
1044 000244 000000 UNIT:      .WORD 0      ; UNIT NUMBER
1045          ;
1046 000246 000000 CLKTAB:   .WORD 0      ;LINE CLOCK STATUS REGISTER
1047 000246 000000 CLKCSR:   .WORD 0      ;LINE CLOCK PRIORITY
1048 000250 000000 CLKBR:    .WORD 0      ;LINE CLOCK VECTOR
1049 000252 000000 CLKVEC:   .WORD 0      ;LINE CLOCK FREQUENCY
1050 000254 000000 CLKFRE:   .WORD 0
1051          ;
1052 000256 000000 DEST:    .WORD 0      ; DESTINATION ADDRESS
1053 000260 000000          .WORD 0
1054 000262 000000          .WORD 0
1055          ;
1056 000264 000000 SRC:      .WORD 0      ; SOURCE ADDRESS
1057 000266 000000          .WORD 0
1058 000270 000000          .WORD 0
1059          ;
1060 000272 000000 DFAULT:   .WORD 0      ; DEFAULT ADDRESS
1061 000274 000010          .WORD 10
1062 000276 000000          .WORD 0
1063          ;
1064          ;DATA STRUCTURES
1065          ;
1066          ;
1067 000300 PCBB:      .BLKW 4      ; PORT CONTROL BLOCK
1068 000310 UDDB:      .BLKW 100.   ; UNIBUS DATA BLOCK
1069 000620 TDRB:      .BLKW 16.     ; TRANSMIT DESCRIPTOR RING
1070 000660 RDRB:      .BLKW 16.     ; RECEIVE DESCRIPTOR RING
1071 000720 RDRBE:     .BLKW 30.     ; EXTENDED TDRB
1072 001014 TDRX:      .BLKW 196.   ; VERY EXTENDED TDRB
1073 001624 RDRX:      .BLKW 196.   ; VERY EXTENDED RDRB
1074          ;
1075 002434 004000 .WORD TEND-TBUF ;LENGTH OF TRANSMIT BUFFERS IN BYTES
1076 002436 TBUF:      .BLKW 128.   ; TRANSMIT BUFFER
1077 003036 TBUF2:     .BLKW 128.
1078 003436 TBUF3:     .BLKW 128.
1079 004036 TBUF4:     .BLKW 128.
1080 004436 TBUF5:     .BLKW 128.
1081 005036 TBUF6:     .BLKW 128.
1082 005436 TBUF7:     .BLKW 128.
1083 006036 TBUF8:     .BLKW 128.
    
```

```

1084      006436'      TEND = .
1085
1086 006436 004000      .WORD  REND-RBUF      ;LENGTH OF RECEIVE BUFFERS IN BYTES
1087 006440      RBUF:      .BLKW  128.      ; RECEIVE BUFFER
1088 007040      RBUF2:     .BLKW  128.
1089 007440      RBUF3:     .BLKW  128.
1090 010040      RBUF4:     .BLKW  128.
1091 010440      RBUF5:     .BLKW  128.
1092 011040      RBUF6:     .BLKW  128.
1093 011440      RBUF7:     .BLKW  128.
1094 012040      RBUF8:     .BLKW  128.
1095      012440'      REND = .
1096
1097
1098      ;DEFAULT PORT FUNCTIONS
1099
1100 012440 000000      NOPF:      .WORD  0      ; NOP FUNCTION
1101 012442 000000      .WORD  0
1102 012444 000000      .WORD  0
1103 012446 000000      .WORD  0
1104
1105 012450 000001      ;LSMA:      .WORD  1      ; LOAD AND START MICROADDRESS FUNCTION
1106 012452 177777      .WORD  177777 ; STARTING INTERNAL ADDRESS OF SELFTEST
1107 012454 000000      .WORD  0
1108 012456 000000      .WORD  0
1109
1110 012460 000002      ;RDDEFA:   .WORD  2      ; READ DEFAULT PHYSICAL ADDRESS FUNCTION
1111 012462 000000      .WORD  0
1112 012464 000000      .WORD  0
1113 012466 000000      .WORD  0
1114
1115 012470 000004      ;RDPHYA:   .WORD  4      ; READ PHYSICAL ADDRESS FUNCTION
1116 012472 000000      .WORD  0
1117 012474 000000      .WORD  0
1118 012476 000000      .WORD  0
1119
1120 012500 000005      ;WTPHYA:   .WORD  5      ; WRITE PHYSICAL ADDRESS
1121 012502 000000      .WORD  0      ; PHYADR
1122 012504 000000      .WORD  0      ; PHYADR
1123 012506 000000      .WORD  0      ; PHYADR
1124
1125 012510 000006      ;RDMULA:   .WORD  6      ; READ MULTICAST ADDRESS LIST FUNCTION
1126 012512 000310'      .WORD  UD88   ; ADDRESS OF UNIBUS DATA BLOCK BASE
1127 012514 005000      .WORD  5000   ; MULTICAST ADDR TABLE LENGTH= 10(10)
1128 012516 000000      .WORD  0
1129
1130 012520 000007      ;WTMULA:   .WORD  7      ; WRITE MULTICAST ADDRESS LIST FUNCTION
1131 012522 000310'      .WORD  UD88   ; ADDRESS OF UNIBUS DATA BLOCK BASE
1132 012524 005000      .WORD  5000   ; MULTICAST ADDR TABLE LENGTH= 10(10)
1133 012526 000000      .WORD  0
1134
1135 012530 000010      ;RDRNGS:   .WORD  10     ; READ RING FORMAT FUNCTION
1136 012532 000310'      .WORD  UD88   ; ADDRESS OF UNIBUS DATA BLOCK BASE
1137 012534 000000      .WORD  0
1138 012536 000000      .WORD  C
1139
1140 012540 000011      ;WTRNGS:   .WORD  11     ; WRITE RING FORMAT FUNCTION
    
```

```

1141 012542 000310' .WORD UDBB ; ADDRESS OF UNIBUS DATA BLOCK BASE
1142 012544 000000 .WORD 0
1143 012546 000000 .WORD 0
1144
1145 012550 000012 ; RDCNT: .WORD 12 ; READ COUNTERS FUNCTION
1146 012552 000310' .WORD UDBB ; ADDRESS OF UNIBUS DATA BLOCK BASE
1147 012554 000000 .WORD 0
1148 012556 000070 .WORD 70 ; COUNTERS LIST LENGTH= 56(10)
1149
1150 012560 000013 ; CLRCNT: .WORD 13 ; READ AND CLEAR COUNTERS FUNCTION
1151 012562 000310' .WORD UDBB ; ADDRESS OF UNIBUS DATA BLOCK BASE
1152 012564 000000 .WORD 0
1153 012566 000070 .WORD 70 ; COUNTERS LIST LENGTH= 56(10)
1154
1155 012570 000014 ; RDMODE: .WORD 14 ; READ MODE FUNCTION
1156 012572 000000 .WORD 0
1157 012574 000000 .WORD 0
1158 012576 000000 .WORD 0
1159
1160 012600 000015 ; WTMODE: .WORD 15 ; WRITE MODE FUNCTION
1161 012602 100104 .WORD 100104 ; PROM AND INTERNAL LOOPBACK MODE
1162 ; GENERATE CRC
1163 012604 000000 .WORD 0
1164 012606 000000 .WORD 0
1165
1166 012610 000015 ; WTMOD1: .WORD 15 ; WRITE MODE FUNCTION
1167 012612 104104 .WORD 104104 ; PROM AND INTERN LOOPBACK AND ENABL COLL TEST
1168 ; GENERATE CRC
1169 012614 000000 .WORD 0
1170 012616 000000 .WORD 0
1171
1172 012620 000015 ; WTMOD2: .WORD 15 ; WRITE MODE FUNCTION
1173 012622 100114 .WORD 100114 ; PROM,INTERNAL LOOPBACK,NO GENERATE CRC
1174 012624 000000 .WORD 0
1175 012626 000000 .WORD 0
1176
1177 012630 000015 ; WTMOD3: .WORD 15 ; WRITE MODE FUNCTION
1178 012632 100004 .WORD 100004 ; PROM,EXT.LOOPBACK, GENERATE CRC
1179 012634 000000 .WORD 0
1180 012636 000000 .WORD 0
1181
1182 012640 000015 ; WTMOD4: .WORD 15 ; WRITE MODE FUNCTION
1183 012642 100014 .WORD 100014 ; PROM;EXT.LOOPBACK, NO GENERATF CRC
1184
1185 012644 000016 ; RDSTA: .WORD 16 ; READ STATUS FUNCTION
1186 012646 000000 .WORD 0
1187 012650 000000 .WORD 0
1188 012652 000000 .WORD 0
1189
1190 012654 000017 ; CLRSTA: .WORD 17 ; READ AND CLEAR STATUS FUNCTION
1191 012656 000000 .WORD 0
1192 012660 000000 .WORD 0
1193 012662 000000 .WORD 0
1194
1195 012664 000020 ; DMPMEM: .WORD 20 ; DUMP INTERNAL MEMORY FUNCTION
1196 012666 000310' .WORD UDBB ; ADDRESS OF UNIBUS DATA BLOCK BASE
1197 012670 000000 .WORD 0
    
```

```

1198 012672 000000 .WORD 0
1199
1200 012674 000021 ; LDMEM: .WORD 21 ; LOAD INTERNAL MEMORY FUNCTION
1201 012676 000310' .WORD UDBB ; ADDRESS OF UNIBUS DATA BLOCK BA'E
1202 012700 000000 .WORD 0
1203 012702 000000 .WORD 0
1204
1205
1206 ;DEFAULT RING FORMATS
1207
1208 012704 000620' RFRMT: .WORD TDRB ; TRANSMIT DESCRIPTOR RING ADDRESS
1209 012706 002000 .WORD 2000 ; TELEN = 6
1210 012710 000004 .WORD 4 ; TRLEN = 4
1211 012712 000660' .WORD RDRB ; RECEIVE DESCRIPTOR RING ADDRESS
1212 012714 002000 .WORD 2000 ; RELEN = 6
1213 012716 000004 .WORD 4 ; RRLEN = 4
1214
1215 012720 001014' RFRMTX: .WORD TDRX ; TRANSMIT DESCRIPTOR RING ADDRESS
1216 012722 002000 .WORD 2000 ; TELEN = 6
1217 012724 000063 .WORD 51. ; TRLEN = 51
1218 012726 000660' .WORD RDRB ; RECEIVE DESCRIPTOR RING ADDRESS
1219 012730 002000 .WORD 2000 ; RELEN = 6
1220 012732 000062 .WORD 50. ; RRLEN = 50
1221
1222
1223 012734 001014' RFRMTE: .WORD TDRX ; TRANSMIT DESCRIPTOR RING ADDRESS
1224 012736 002000 .WORD 2000 ; TELEN = 6
1225 012740 000063 .WORD 51. ; TRLEN = 51
1226 012742 001624' .WORD RDRX ; RECEIVE DESCRIPTOR RING ADDRESS
1227 012744 002000 .WORD 2000 ; RELEN = 6
1228 012746 000062 .WORD 50. ; RRLEN = 3C
1229
1230
1231 ;DEFAULT RECEIVE DESCRIPTOR RINGS
1232
1233 012750 000040 RDRB1A: .WORD 32. ; SLEN = 32 BYTES
1234 012752 006440' .WORD RBUF ; SEGB = RBUF
1235 012754 100000 .WORD 100000 ; OWN = UNA
1236 012756 000000 .WORD 0
1237
1238 012760 000040 ; .WORD 32. ; SLEN = 32 BYTES
1239 012762 006440' .WORD RBUF ; SEGB = RBUF
1240 012764 000000 .WORD 0 ; OWN = PORT DRIVER
1241 012766 000000 .WORD 0
1242
1243 012770 000040 ; .WORD 32. ; SLEN = 32 BYTES
1244 012772 006440' .WORD RBUF ; SEGB = RBUF
1245 012774 000000 .WORD 0 ; OWN = PORT DRIVER
1246 012776 000000 .WORD 0
1247
1248 013000 000040 ; .WORD 32. ; SLEN = 32 BYTES
1249 013002 006440' .WORD RBUF ; SEGB = RBUF
1250 013004 000000 .WORD 0 ; OWN = PORT DRIVER
1251 013006 000000 .WORD 0
1252
1253 013010 000040 ; RDRB1B: .WORD 32. ; SLEN 32 BYTES
1254 013012 006440' .WORD RBUF ; SEGB = RBUF
    
```

```

1255 013014 000000 .WORD 0 ; OWN = PORT DRIVER
1256 013016 000000 .WORD 0
1257 ;
1258 013020 000040 .WORD 32. ; SLEN = 32 BYTES
1259 013022 006440 .WORD RBUF ; SEGB = RBUF
1260 013024 000000 .WORD 0 ; OWN = PORT DRIVER
1261 013026 000000 .WORD 0
1262 ;
1263 013030 000040 .WORD 32. ; SLEN = 32 BYTES
1264 013032 006440 .WORD RBUF ; SEGB = RBUF
1265 013034 000000 .WORD 0 ; OWN = PORT DRIVER
1266 013036 000000 .WORD 0
1267 ;
1268 013040 000040 .WORD 32. ; SLEN = 32 BYTES
1269 013042 006440 .WORD RBUF ; SEGB = RBUF
1270 013044 000000 .WORD 0 ; OWN = PORT DRIVER
1271 013046 000000 .WORD 0
1272 ;
1273 013050 000040 RDRB2A: .WORD 32. ; SLEN = 32 BYTES
1274 013052 006440 .WORD RBUF ; SEGB = RBUF
1275 013054 100000 .WORD 100000 ; OWN = UNA
1276 013056 000000 .WORD 0
1277 ;
1278 013060 000040 .WORD 32. ; SLEN = 32 BYTES
1279 013062 006440 .WORD RBUF ; SEGB = RBUF
1280 013064 100000 .WORD 100000 ; OWN = UNA
1281 013066 000000 .WORD 0
1282 ;
1283 013070 000040 .WORD 32. ; SLEN = 32 BYTES
1284 013072 006440 .WORD RBUF ; SEGB = RBUF
1285 013074 000000 .WORD 0 ; OWN = PORT DRIVER
1286 013076 000000 .WORD 0
1287 ;
1288 013100 000040 .WORD 32. ; SLEN = 32 BYTES
1289 013102 006440 .WORD RBUF ; SEGB = RBUF
1290 013104 000000 .WORD 0 ; OWN = PORT DRIVER
1291 013106 000000 .WORD 0
1292 ;
1293 013110 000100 RDRB3A: .WORD 64. ; SLEN = 64 BYTES
1294 013112 006440 .WORD RBUF ; SEGB = RBUF
1295 013114 100000 .WORD 100000 ; OWN = LUA
1296 013116 000000 .WORD 0
1297 ;
1298 013120 000100 .WORD 64. ; SLEN = 64 BYTES
1299 013122 007040 .WORD RBUF2 ; SEGB = RBUF2
1300 013124 100000 .WORD 100000 ; OWN = LUA
1301 013126 000000 .WORD 0
1302 ;
1303 013130 000100 .WORD 64. ; SLEN = 64 BYTES
1304 013132 007440 .WORD RBUF3 ; SEGB = RBUF3
1305 013134 100000 .WORD 100000 ; OWN = LUA
1306 013136 000000 .WORD 0
1307 ;
1308 013140 000100 .WORD 64. ; SLEN = 64 BYTES
1309 013142 006440 .WORD RBUF ; SEGB = RBUF
1310 013144 000000 .WORD 0 ; OWN = PORT DRIVER
1311 013146 000000 .WORD 0
    
```

```

1312
1313 ;
1314 013150 000040 ;RDR848: .WORD 32. ; SLEN = 32 BYTES
1315 013152 006440' .WORD RBUF ; SEGB = RBUF
1316 013154 100000 .WORD 100000 ; OWN = LUA
1317 013156 000000 .WORD 0 ;
1318 ;
1319 013160 000040 .WORD 32. ;
1320 013162 006440' .WORD RBUF ;
1321 013164 100000 .WORD 100000 ;
1322 013166 000000 .WORD 0 ;
1323 ;
1324 013170 000040 .WORD 32. ;
1325 013172 006440' .WORD RBUF ;
1326 013174 100000 .WORD 100000 ;
1327 013176 000000 .WORD 0 ;
1328 ;
1329 013200 000040 .WORD 32. ;
1330 013202 006440' .WORD RBUF ;
1331 013204 100000 .WORD 100000 ;
1332 013206 000000 .WORD 0 ;
1333 ;
1334 013210 000040 .WORD 32. ;
1335 013212 006440' .WORD RBUF ;
1336 013214 100000 .WORD 100000 ;
1337 013216 000000 .WORD 0 ;
1338 ;
1339 013220 000040 .WORD 32. ; SLEN = 32 BYTES
1340 013222 006440' .WORD RBUF ; SEGB = RBUF
1341 013224 100000 .WORD 100000 ; OWN = LUA
1342 013226 000000 .WORD 0 ;
1343 ;
1344 013230 000040 .WORD 32. ;
1345 013232 006440' .WORD RBUF ;
1346 013234 100000 .WORD 100000 ;
1347 013236 000000 .WORD 0 ;
1348 ;
1349 013240 000040 .WORD 32. ;
1350 013242 006440' .WORD RBUF ;
1351 013244 100000 .WORD 100000 ;
1352 013246 000000 .WORD 0 ;
1353 ;
1354 013250 000040 .WORD 32. ;
1355 013252 006440' .WORD RBUF ;
1356 013254 100000 .WORD 100000 ;
1357 013256 000000 .WORD 0 ;
1358 ;
1359 013260 000040 .WORD 32. ;
1360 013262 006440' .WORD RBUF ;
1361 013264 100000 .WORD 100000 ;
1362 013266 000000 .WORD 0 ;
1363 ;
1364 013270 000040 .WORD 32. ; SLEN = 32 BYTES
1365 013272 006440' .WORD RBUF ; SEGB = RBUF
1366 013274 100000 .WORD 100000 ; OWN = LUA
1367 013276 000000 .WORD 0 ;
1368 ;
    
```



1369	013300	000040	.WORD	32.	;
1370	013302	006440'	.WORD	RBUF	;
1371	013304	100000	.WORD	100000	;
1372	013306	000000	.WORD	0	;
1373					;
1374	013310	000040	.WORD	32.	;
1375	013312	006440'	.WORD	RBUF	;
1376	013314	100000	.WORD	100000	;
1377	013316	000000	.WORD	0	;
1378					;
1379	013320	000040	.WORD	32.	;
1380	013322	006440'	.WORD	RBUF	;
1381	013324	100000	.WORD	100000	;
1382	013326	000000	.WORD	0	;
1383					;
1384	013330	000040	.WORD	32.	;
1385	013332	006440'	.WORD	RBUF	;
1386	013334	100000	.WORD	100000	;
1387	013336	000000	.WORD	0	;
1388					;
1389	013340	000040	.WORD	32.	;
1390	013342	006440'	.WORD	RBUF	;
1391	013344	100000	.WORD	100000	;
1392	013346	000000	.WORD	0	;
1393					;
1394	013350	000040	.WORD	32.	;
1395	013352	006440'	.WORD	RBUF	;
1396	013354	100000	.WORD	100000	;
1397	013356	000000	.WORD	0	;
1398					;
1399	013360	000040	.WORD	32.	;
1400	013362	006440'	.WORD	RBUF	;
1401	013364	100000	.WORD	100000	;
1402	013366	000000	.WORD	0	;
1403					;
1404	013370	000040	.WORD	32.	;
1405	013372	006440'	.WORD	RBUF	;
1406	013374	100000	.WORD	100000	;
1407	013376	000000	.WORD	0	;
1408					;
1409	013400	000040	.WORD	32.	;
1410	013402	006440'	.WORD	RBUF	;
1411	013404	100000	.WORD	100000	;
1412	013406	000000	.WORD	0	;
1413					;
1414					;
1415	013410	000040	.WORD	32.	;
1416	013412	006440'	.WORD	RBUF	;
1417	013414	100000	.WORD	100000	;
1418	013416	000000	.WORD	0	;
1419					;
1420	013420	000040	.WORD	32.	;
1421	013422	006440'	.WORD	RBUF	;
1422	013424	100000	.WORD	100000	;
1423	013426	000000	.WORD	0	;
1424					;
1425	013430	000040	.WORD	32.	;

RDRB5A:

SLEN = 32 BYTES  
 SEGB = RBUF  
 OWN = LUA

SLEN = 32 BYTES  
 SEGB = RBUF  
 OWN = LUA

SLEN = 32 BYTES  
 SEGB = RBUF  
 OWN = LUA

SLEN = 32 BYTES

1426	013432	006440'	.WORD	RBUF	; SEGB = RBUF
1427	013434	100000	.WORD	100000	; OWN = LUA
1428	013436	000000	.WORD	0	;
1429					;
1430	013440	000040	.WORD	32.	; SLEN = 32 BYTES
1431	013442	006440'	.WORD	RBUF	; SEGB = RBUF
1432	013444	100000	.WORD	100000	; OWN = LUA
1433	013446	000000	.WORD	0	;
1434					;
1435	013450	000040	.WORD	32.	; SLEN = 32 BYTES
1436	013452	006440'	.WORD	RBUF	; SEGB = RBUF
1437	013454	100000	.WORD	100000	; OWN = LUA
1438	013456	000000	.WORD	0	;
1439					;
1440	013460	000040	.WORD	32.	; SLEN = 32 BYTES
1441	013462	006440'	.WORD	RBUF	; SEGB = RBUF
1442	013464	100000	.WORD	100000	; OWN = LUA
1443	013466	000000	.WORD	0	;
1444					;
1445	013470	000040	.WORD	32.	; SLEN = 32 BYTES
1446	013472	006440'	.WORD	RBUF	; SEGB = RBUF
1447	013474	100000	.WORD	100000	; OWN = LUA
1448	013476	000000	.WORD	0	;
1449					;
1450	013500	000040	.WORD	32.	; SLEN = 32 BYTES
1451	013502	006440'	.WORD	RBUF	; SEGB = RBUF
1452	013504	100000	.WORD	100000	; OWN = LUA
1453	013506	000000	.WORD	0	;
1454					;
1455	013510	000040	.WORD	32.	; SLEN = 32 BYTES
1456	013512	006440'	.WORD	RBUF	; SEGB = RBUF
1457	013514	100000	.WORD	100000	; OWN = LUA
1458	013516	000000	.WORD	0	;
1459					;
1460	013520	000040	.WORD	32.	; SLEN = 32 BYTES
1461	013522	006440'	.WORD	RBUF	; SEGB = RBUF
1462	013524	100000	.WORD	100000	; OWN = LUA
1463	013526	000000	.WORD	0	;
1464					;
1465	013530	000040	.WORD	32.	; SLEN = 32 BYTES
1466	013532	006440'	.WORD	RBUF	; SEGB = RBUF
1467	013534	000000	.WORD	0	; OWN = PORT DRIVER
1468	013536	000000	.WORD	0	;
1469					;
1470					;
1471	013540	000020	.WORD	16.	; SLEN = 16 BYTES
1472	013542	006440'	.WORD	RBUF	; SEGB = RBUF
1473	013544	100000	.WORD	100000	; OWN = LUA
1474	013546	000000	.WORD	0	;
1475					;
1476	013550	000026	.WORD	22.	; SLEN = 22 BYTES (INCL. CRC)
1477	013552	007040'	.WORD	RBUF2	; SEGB = RBUF2
1478	013554	100000	.WORD	100000	; OWN = LUA
1479	013556	000000	.WORD	0	;
1480					;
1481	013560	000020	.WORD	16.	; SLEN = 16 BYTES
1482	013562	006440'	.WORD	RBUF	; SEGB = RBUF

RDRB4A:

```

1483 013564 000000      .WORD 000000 ; OWN = PORT DRIVER
1484 013566 000000      .WORD 0 ;
1485                    ;
1486                    ;
1487 013570 000040      .WORD 32. ; SLEN = 32 BYTES
1488 013572 006440      .WORD RBUF ; SEGB = RBUF
1489 013574 100000      .WORD 100000 ; OWN = LUA
1490 013576 000000      .WORD 0 ;
1491
1492 013600 000040      .WORD 32. ;
1493 013602 006440      .WORD RBUF ;
1494 013604 100000      .WORD 100000 ;
1495 013606 000000      .WORD 0 ;
1496
1497 013610 000040      .WORD 32. ;
1498 013612 006440      .WORD RBUF ;
1499 013614 100000      .WORD 100000 ;
1500 013616 000000      .WORD 0 ;
1501
1502 013620 000040      .WORD 32. ;
1503 013622 006440      .WORD RBUF ;
1504 013624 100000      .WORD 100000 ;
1505 013626 000000      .WORD 0 ;
1506
1507 013630 000040      .WORD 32. ; SLEN = 32 BYTES
1508 013632 006440      .WORD RBUF ; SEGB = RBUF
1509 013634 100000      .WORD 100000 ; OWN = LUA
1510 013636 000000      .WORD 0 ;
1511
1512 013640 000040      .WORD 32. ;
1513 013642 006440      .WORD RBUF ;
1514 013644 100000      .WORD 100000 ;
1515 013646 000000      .WORD 0 ;
1516
1517 013650 000040      .WORD 32. ;
1518 013652 006440      .WORD RBUF ;
1519 013654 100000      .WORD 100000 ;
1520 013656 000000      .WORD 0 ;
1521
1522 013660 000040      .WORD 32. ;
1523 013662 006440      .WORD RBUF ;
1524 013664 100000      .WORD 100000 ;
1525 013666 000000      .WORD 0 ;
1526
1527 013670 000040      .WORD 32. ;
1528 013672 006440      .WORD RBUF ;
1529 013674 100000      .WORD 100000 ;
1530 013676 000000      .WORD 0 ;
1531
1532 013700 000040      .WORD 32. ;
1533 013702 006440      .WORD RBUF ;
1534 013704 100000      .WORD 100000 ;
1535 013706 000000      .WORD 0 ;
1536
1537 013710 000040      .WORD 32. ; SLEN = 32 BYTES
1538 013712 006440      .WORD RBUF ; SEGB = RBUF
1539 013714 100000      .WORD 100000 ; OWN = LUA
    
```

1540	013716	000000	.WORD	0	;
1541					
1542	013720	000040	.WORD	32.	;
1543	013722	006440	.WORD	RBUF	;
1544	013724	100000	.WORD	100000	;
1545	013726	000000	.WORD	0	;
1546					
1547	013730	000040	.WORD	32.	;
1548	013732	006440	.WORD	RBUF	;
1549	013734	100000	.WORD	100000	;
1550	013736	000000	.WORD	0	;
1551					
1552	013740	000040	.WORD	32.	;
1553	013742	006440	.WORD	RBUF	;
1554	013744	100000	.WORD	100000	;
1555	013746	000000	.WORD	0	;
1556					
1557	013750	000040	.WORD	32.	;
1558	013752	006440	.WORD	RBUF	;
1559	013754	100000	.WORD	100000	;
1560	013756	000000	.WORD	0	;
1561					
1562	013760	000040	.WORD	32.	;
1563	013762	006440	.WORD	RBUF	;
1564	013764	100000	.WORD	100000	;
1565	013766	000000	.WORD	0	;
1566					
1567	013770	000040	.WORD	32.	;
1568	013772	006440	.WORD	RBUF	;
1569	013774	100000	.WORD	100000	;
1570	013776	000000	.WORD	0	;
1571					
1572	014000	000040	.WORD	32.	;
1573	014002	006440	.WORD	RBUF	;
1574	014004	100000	.WORD	100000	;
1575	014006	000000	.WORD	0	;
1576					
1577	014010	000040	.WORD	32.	;
1578	014012	006440	.WORD	RBUF	;
1579	014014	100000	.WORD	100000	;
1580	014016	000000	.WORD	0	;
1581					
1582	014020	000040	.WORD	32.	;
1583	014022	006440	.WORD	RBUF	;
1584	014024	100000	.WORD	100000	;
1585	014026	000000	.WORD	0	;
1586					
1587	014030	000040	.WORD	32.	;
1588	014032	006440	.WORD	RBUF	;
1589	014034	100000	.WORD	100000	;
1590	014036	000000	.WORD	0	;
1591					
1592	014040	000040	.WORD	32.	;
1593	014042	006440	.WORD	RBUF	;
1594	014044	100000	.WORD	100000	;
1595	014046	000000	.WORD	0	;
1596					

; SLEN = 32 BYTES  
 ; SEGB = RBUF  
 ; OWN = LUA

; SLEN = 32 BYTES  
 ; SEGB = RBUF  
 ; OWN = DELUA

; SLEN = 32 BYTES  
 ; SEGB = RBUF  
 ; OWN = LUA

1597	014050	000040	.WORD	32.	;
1598	014052	006440'	.WORD	RBUF	;
1599	014054	100000	.WORD	100000	;
1600	014056	000000	.WORD	0	;
1601					
1602	014060	000040	.WORD	32.	;
1603	014062	006440'	.WORD	RBUF	;
1604	014064	100000	.WORD	100000	;
1605	014066	000000	.WORD	0	;
1606					
1607	014070	000040	.WORD	32.	;
1608	014072	006440'	.WORD	RBUF	;
1609	014074	100000	.WORD	100000	;
1610	014076	000000	.WORD	0	;
1611					
1612	014100	000040	.WORD	32.	; SLEN = 32 BYTES
1613	014102	006440'	.WORD	RBUF	; SEGB = RBUF
1614	014104	100000	.WORD	100000	; OWN = LUA
1615	014106	000000	.WORD	0	;
1616					
1617	014110	000040	.WORD	32.	;
1618	014112	006440'	.WORD	RBUF	;
1619	014114	100000	.WORD	100000	;
1620	014116	000000	.WORD	0	;
1621					
1622	014120	000040	.WORD	32.	;
1623	014122	006440'	.WORD	RBUF	;
1624	014124	100000	.WORD	100000	;
1625	014126	000000	.WORD	0	;
1626					
1627	014130	000040	.WORD	32.	;
1628	014132	006440'	.WORD	RBUF	;
1629	014134	100000	.WORD	100000	;
1630	014136	000000	.word	0	;
1631					
1632	014140	000040	.WORD	32.	;
1633	014142	006440'	.WORD	RBUF	;
1634	014144	100000	.WORD	100000	;
1635	014146	000000	.WORD	0	;
1636					
1637	014150	000040	.WORD	32.	; SLEN = 32 BYTES
1638	014152	006440'	.WORD	RBUF	; SEGB = RBUF
1639	014154	100000	.WORD	100000	; OWN = LUA
1640	014156	000000	.WORD	0	;
1641					
1642	014160	000040	.WORD	32.	;
1643	014162	006440'	.WORD	RBUF	;
1644	014164	100000	.WORD	100000	;
1645	014166	000000	.WORD	0	;
1646					
1647	014170	000040	.WORD	32.	;
1648	014172	006440'	.WORD	RBUF	;
1649	014174	100000	.WORD	100000	;
1650	014176	000000	.WORD	0	;
1651					
1652	014200	000040	.WORD	32.	;
1653	014202	006440'	.WORD	RBUF	;

1654	014204	100000	.WORD	100000	;
1655	014206	000000	.WORD	0	;
1656					
1657	014210	000040	.WORD	32.	;
1658	014212	006440	.WORD	RBUF	;
1659	014214	100000	.WORD	100000	;
1660	014216	000000	.WORD	0	;
1661					
1662	014220	000040	.WORD	32.	; SLEN = 32 BYTES
1663	014222	006440	.WORD	RBUF	; SEGB = RBUF
1664	014224	100000	.WORD	100000	; OWN = LUA
1665	014226	000000	.WORD	0	;
1666					
1667	014230	000040	.WORD	32.	;
1668	014232	006440	.WORD	RBUF	;
1669	014234	100000	.WORD	100000	;
1670	014236	000000	.WORD	0	;
1671					
1672	014240	000040	.WORD	32.	;
1673	014242	006440	.WORD	RBUF	;
1674	014244	100000	.WORD	100000	;
1675	014246	000000	.WORD	0	;
1676					
1677	014250	000040	.WORD	32.	;
1678	014252	006440	.WORD	RBUF	;
1679	014254	100000	.WORD	100000	;
1680	014256	000000	.WORD	0	;
1681					
1682	014260	000040	.WORD	32.	;
1683	014262	006440	.WORD	RBUF	;
1684	014264	100000	.WORD	100000	;
1685	014266	000000	.WORD	0	;
1686					
1687					
1688	014270	000040	.WORD	32.	; SLEN = 32 BYTES
1689	014272	006440	.WORD	RBUF	; SEGB = RBUF
1690	014274	100000	.WORD	100000	; OWN = LUA
1691	014276	000000	.WORD	0	;
1692					
1693	014300	000040	.WORD	32.	;
1694	014302	006440	.WORD	RBUF	;
1695	014304	100000	.WORD	100000	;
1696	014306	000000	.WORD	0	;
1697					
1698	014310	000040	.WORD	32.	;
1699	014312	006440	.WORD	RBUF	;
1700	014314	100000	.WORD	100000	;
1701	014316	000000	.WORD	0	;
1702					
1703	014320	000040	.WORD	32.	;
1704	014322	006440	.WORD	RBUF	;
1705	014324	100000	.WORD	100000	;
1706	014326	000000	.WORD	0	;
1707					
1708	014330	000040	.WORD	32.	;
1709	014332	006440	.WORD	RBUF	;
1710	014334	100000	.WORD	100000	;

```

1711 014336 000000 .WORD 0 ;
1712
1713 014340 000040 .WORD 32. ; SLEN = 32 BYTES
1714 014342 006440' .WORD RBUF ; SEGB = RBUF
1715 014344 100000 .WORD 100000 ; OWN = LUA
1716 014346 000000 .WORD 0 ;
1717
1718 014350 000040 .WORD 32. ;
1719 014352 006440' .WORD RBUF ;
1720 014354 100000 .WORD 100000 ;
1721 014356 000000 .WORD 0 ;
1722
1723 014360 000040 .WORD 32. ;
1724 014362 006440' .WORD RBUF ;
1725 014364 100000 .WORD 100000 ;
1726 014366 000000 .WORD 0 ;
1727
1728 014370 000040 .WORD 32. ;
1729 014372 006440' .WORD RBUF ;
1730 014374 000000 .WORD 0 ; OWN = PORT DRIVER
1731 014376 000000 .WORD 0 ;
1732
1733 014400 000040 .WORD 32. ;
1734 014402 006440' .WORD RBUF ;
1735 014404 000000 .WORD 0 ; OWN = PORT DRIVER
1736 014406 000000 .WORD 0 ;
1737
1738
1739 ;
1740 ;DEFAULT TRANSMIT DESCRIPTOR RINGS
1741 ;
1742 014410 000032 TDRB1A: .WORD 26. ; SLEN = 26 BYTES
1743 014412 002436' .WORD TBUF ; SEGB = TBUF
1744 014414 101400 .WORD 101400 ; OWN = UNA ;STP,ENP
1745 014416 000000 .WORD 0 ;
1746 ;
1747 014420 000030 .WORD 24. ; SLEN = 24 BYTES
1748 014422 002436' .WORD TBUF ; SEGB = TBUF
1749 014424 000000 .WORD 0 ; OWN = PORT DRIVER
1750 014426 000000 .WORD 0 ;
1751 ;
1752 014430 000030 .WORD 24. ; SLEN = 24 BYTES
1753 014432 002436' .WORD TBUF ; SEGB = TBUF
1754 014434 000000 .WORD 0 ; OWN = PORT DRIVER
1755 014436 000000 .WORD 0 ;
1756 ;
1757 014440 000030 .WORD 24. ; SLEN = 24 BYTES
1758 014442 002436' .WORD TBUF ; SEGB = TBUF
1759 014444 000000 .WORD 0 ; OWN = PORT DRIVER
1760 014446 000000 .WORD 0 ;
1761 ;
1762 014450 000040 TDRB1B: .WORD 32. ; SLEN = 32 BYTES
1763 014452 002436' .WORD TBUF ; SEGB = TBUF
1764 014454 101400 .WORD 101400 ; OWN = UNA ;STP,ENP
1765 014456 000000 .WORD 0 ;
1766 ;
1767 014460 000040 .WORD 32. ; SLEN = 32 BYTES
    
```

```

1768 014462 002436' .WORD TBUF ; SEGB = TBUF
1769 014464 000000 .WORD 0 ; OWN = PORT DRIVER
1770 014466 000000 .WORD 0
1771 ;
1772 014470 000040 .WORD 32. ; SLEN = 32 BYTES
1773 014472 002436' .WORD TBUF ; SEGB = TBUF
1774 014474 000000 .WORD 0 ; OWN = PORT DRIVER
1775 014476 000000 .WORD 0
1776 ;
1777 014500 000040 .WORD 32. ; SLEN = 32 BYTES
1778 014502 002436' .WORD TBUF ; SEGB = TBUF
1779 014504 000000 .WORD 0 ; OWN = PORT DRIVER
1780 014506 000000 .WORD 0
1781 ;
1782 014510 TDRB1C: .WORD 34. ; SLEN = 34 BYTES
1783 014510 000042 .WORD TBUF ; SEGB = TBUF
1784 014512 002436' .WORD 101400 ; OWN = LUA;STP;ENP
1785 014514 101400 .WORD 0 ;
1786 014516 000000 .WORD 0
1787 ;
1788 014520 000042 .WORD 34. ; SLEN = 34 BYTES
1789 014522 002436' .WORD TBUF ; SEGB = TBUF
1790 014524 001400 .WORD 001400 ; OWN = PORT DRIVER;STP;ENP
1791 014526 000000 .WORD 0 ;
1792 ;
1793 014530 TDRB1D: .WORD 40. ; SLEN = 40 BYTES
1794 014532 002436' .WORD TBUF ; SEGB = TBUF
1795 014534 101400 .WORD 101400 ; OWN = LUA;STP;ENP
1796 014536 000000 .WORD 0 ;
1797 ;
1798 014540 000050 .WORD 40. ; SLEN = 40 BYTES
1799 014542 002436' .WORD TBUF ; SEGB = TBUF
1800 014544 001400 .WORD 001400 ; OWN = PORT DRIVER;STP;ENP
1801 014546 000000 .WORD 0 ;
1802 ;
1803 01455 TDRB1E: .WORD 14. ; SLEN = 14 BYTES
1804 014550 000016 .WORD TBUF ; SEGB = TBUF
1805 014552 002436' .WORD 101000 ; OWN = DELUA ;STP
1806 014554 101000 .WORD 0 ;
1807 014556 000000 .WORD 0
1808 ;
1809 014560 000022 .WORD 18. ; SLEN = 18 BYTES
1810 014562 002436' .WORD TBUF ; SEGB = TBUF
1811 014564 101000 .WORD 101000 ; OWN = DELUA ;STP
1812 014566 000000 .WORD 0 ;
1813 ;
1814 014570 000022 .WORD 18. ; SLEN = 18 BYTES
1815 014572 002436' .WORD TBUF ; SEGB = TBUF
1816 014574 000000 .WORD 0 ; OWN = PRT DRIVER
1817 014576 000000 .WORD 0 ;
1818 ;
1819 ;
1820 014600 TDRB2A: .WORD 14. ; SLEN = 14 BYTES
1821 014602 002436' .WORD TBUF ; SEGB = TBUF
1822 014604 101000 .WORD 101000 ; OWN = DELUA ;STP
1823 014606 000000 .WORD 0
1824 ;
    
```



1825	014610	000022	.WORD	18.	; SLEN = 18 BYTES
1826	014612	002436'	.WORD	TBUF	; SEGB = TBUF
1827	014614	100400	.WORD	100400	; OWN = DELUA ;ENP
1828	014616	000000	.WORD	0	
1829					
1830	014620	000020	.WORD	16.	; SLEN = 16 BYTES
1831	014622	002436'	.WORD	TBUF	; SEGB = TBUF
1832	014624	000000	.WORD	0	; OWN = PORT DRIVER
1833	014626	000000	.WORD	0	
1834					
1835	014630	000020	.WORD	16.	; SLEN = 16 BYTES
1836	014632	002436'	.WORD	TBUF	; SEGB = TBUF
1837	014634	000000	.WORD	0	; OWN = PORT DRIVER
1838	014636	000000	.WORD	0	
1839					
1840					
1841	014640	000020	.WORD	20	; SLEN = 20 BYTES
1842	014642	002436'	.WORD	TBUF	; SEGB = TBUF
1843	014644	101000	.WORD	101000	; OWN = UNA;STP
1844	014646	000000	.WORD	0	
1845					
1846	014650	000020	.WORD	20	; SLEN = 20 BYTES
1847	014652	002436'	.WORD	TBUF	; SEGB = TBUF
1848	014654	100400	.WORD	100400	; OWN = UNA;ENP
1849	014656	000000	.WORD	0	
1850					
1851	014660	000020	.WORD	20	; SLEN = 20 BYTES
1852	014662	002436'	.WORD	TBUF	; SEGB = TBUF
1853	014664	101000	.WORD	101000	; OWN = UNA;STP
1854	014666	000000	.WORD	0	
1855					
1856	014670	000020	.WORD	20	; SLEN = 20 BYTES
1857	014672	002436'	.WORD	TBUF	; SEGB = TBUF
1858	014674	100400	.WORD	100400	; OWN = UNA;ENP
1859	014676	000000	.WORD	0	
1860					
1861					
1862					
1863	014700	000040	.WORD	32.	; SLEN = 42 BYTES
1864	014702	002436'	.WORD	TBUF	; SEGB = TBUF
1865	014704	101400	.WORD	101400	; OWN = LUA ;STP,ENP
1866	014706	000000	.WORD	0	
1867					
1868	014710	000042	.WORD	42	; SLEN = 42 BYTES
1869	014712	002436'	.WORD	TBUF	; SEGB = TBUF
1870	014714	100000	.WORD	100000	; OWN = LUA
1871	014716	000000	.WORD	0	
1872					
1873	014720	000052	.WORD	42.	; SLEN = 42 BYTES
1874	014722	002436'	.WORD	TBUF	; SEGB = TBUF3
1875	014724	100400	.WORD	100400	; OWN = LUA ;ENP
1876	014726	000000	.WORD	0	
1877					
1878	014730	000174	.WORD	124.	; SLEN = 124 BYTES
1879	014732	002436'	.WORD	TBUF	; SEGB = TBUF
1880	014734	000000	.WORD	0	; OWN = PORT DRIVER
1881	014736	000000	.WORD	0	

TDRB2B:

TDRB3A:

```

1882 ;
1883 ;
1884 014740 000032 TDRBXX: .WORD 26. ; SLEN = 32 BYTES
1885 014742 002436' .WORD TBUF ; SEGB = TBUF
1886 014744 101400 .WORD 101400 ; OWN = LUA ;STP ;ENP
1887 014746 000000 .WORD 0
1888 ;
1889 014750 000032 .WORD 26. ; SLEN = 32 BYTES
1890 014752 002436' .WORD TBUF ; SEGB = TBUF
1891 014754 101400 .WORD 101400 ; OWN = LUA ;STP ;ENP
1892 014756 000000 .WORD 0
1893 ;
1894 014760 000032 .WORD 26. ; SLEN = 32 BYTES
1895 014762 002436' .WORD TBUF ; SEGB = TBUF
1896 014764 101400 .WORD 101400 ; OWN = LUA ;STP ;ENP
1897 014766 000000 .WORD 0
1898 ;
1899 014770 000032 .WORD 26. ; SLEN = 32 BYTES
1900 014772 002436' .WORD TBUF ; SEGB = TBUF
1901 014774 101400 .WORD 101400 ; OWN = LUA ;STP ;ENP
1902 014776 000000 .WORD 0
1903 ;
1904 015000 000032 .WORD 26. ; SLEN = 32 BYTES
1905 015002 002436' .WORD TBUF ; SEGB = TBUF
1906 015004 101400 .WORD 101400 ; OWN = LUA ;STP ;ENP
1907 015006 000000 .WORD 0
1908 ;
1909 015010 000032 .WORD 26. ; SLEN = 32 BYTES
1910 015012 002436' .WORD TBUF ; SEGB = TBUF
1911 015014 101400 .WORD 101400 ; OWN = LUA ;STP ;ENP
1912 015016 000000 .WORD 0
1913 ;
1914 015020 000032 .WORD 26. ; SLEN = 32 BYTES
1915 015022 002436' .WORD TBUF ; SEGB = TBUF
1916 015024 101400 .WORD 101400 ; OWN = LUA ;STP ;ENP
1917 015026 000000 .WORD 0
1918 ;
1919 015030 000032 .WORD 26. ; SLEN = 32 BYTES
1920 015032 002436' .WORD TBUF ; SEGB = TBUF
1921 015034 101400 .WORD 101400 ; OWN = LUA ;STP ;ENP
1922 015036 000000 .WORD 0
1923 ;
1924 015040 000032 .WORD 26. ; SLEN = 32 BYTES
1925 015042 002436' .WORD TBUF ; SEGB = TBUF
1926 015044 101400 .WORD 101400 ; OWN = LUA ;STP ;ENP
1927 015046 000000 .WORD 0
1928 ;
1929 015050 000032 .WORD 26. ; SLEN = 32 BYTES
1930 015052 002436' .WORD TBUF ; SEGB = TBUF
1931 015054 101400 .WORD 101400 ; OWN = LUA ;STP ;ENP
1932 015056 000000 .WORD 0
1933 ;
1934 015060 000032 .WORD 26. ; SLEN = 32 BYTES
1935 015062 002436' .WORD TBUF ; SEGB = TBUF
1936 015064 101400 .WORD 101400 ; OWN = LUA ;STP ;ENP
1937 015066 000000 .WORD 0
1938 ;
    
```

1939	015070	000032	.WORD	26.	; SLEN = 32 BYTES
1940	015072	002436	.WORD	TBUF	; SEGB = TBUF
1941	015074	101400	.WORD	101400	; OWN = LUA ;STP ;ENP
1942	015076	000000	.WORD	0	
1943			:		
1944	015100	000032	.WORD	26.	; SLEN = 32 BYTES
1945	015102	002436	.WORD	TBUF	; SEGB = TBUF
1946	015104	101400	.WORD	101400	; OWN = LUA ;STP ;ENP
1947	015106	000000	.WORD	0	
1948			:		
1949	015110	000032	.WORD	26.	; SLEN = 32 BYTES
1950	015112	002436	.WORD	TBUF	; SEGB = TBUF
1951	015114	101400	.WORD	101400	; OWN = LUA ;STP ;ENP
1952	015116	000000	.WORD	0	
1953			:		
1954	015120	000032	.WORD	26.	; SLEN = 32 BYTES
1955	015122	002436	.WORD	TBUF	; SEGB = TBUF
1956	015124	101400	.WORD	101400	; OWN = LUA ;STP ;ENP
1957	015126	000000	.WORD	0	
1958			:		
1959	015130	000032	.WORD	26.	; SLEN = 32 BYTES
1960	015132	002436	.WORD	TBUF	; SEGB = TBUF
1961	015134	101400	.WORD	101400	; OWN = LUA ;STP ;ENP
1962	015136	000000	.WORD	0	
1963			:		
1964	015140	000032	.WORD	26.	; SLEN = 32 BYTES
1965	015142	002436	.WORD	TBUF	; SEGB = TBUF
1966	015144	101400	.WORD	101400	; OWN = LUA;STP;ENP
1967	015146	000000	.WORD	0	
1968			:		
1969	015150	000032	.WORD	26.	; SLEN = 32 BYTES
1970	015152	002436	.WORD	TBUF	; SEGB = TBUF
1971	015154	101400	.WORD	101400	; OWN = LUA;STP;ENP
1972	015156	000000	.WORD	0	
1973			:		
1974	015160	000032	.WORD	26.	; SLEN = 32 BYTES
1975	015162	002436	.WORD	TBUF	; SEGB = TBUF
1976	015164	101400	.WORD	101400	; OWN = LUA;STP;ENP
1977	015166	000000	.WORD	0	
1978			:		
1979	015170	000032	.WORD	26.	; SLEN = 32 BYTES
1980	015172	002436	.WORD	TBUF	; SEGB = TBUF
1981	015174	101400	.WORD	101400	; OWN = LUA;STP;ENP
1982	015176	000000	.WORD	0	
1983			:		
1984	015200	000032	.WORD	26.	; SLEN = 32 BYTES
1985	015202	002436	.WORD	TBUF	; SEGB = TBUF
1986	015204	101400	.WORD	101400	; OWN = LUA;STP;ENP
1987	015206	000000	.WORD	0	
1988			:		
1989	015210	000032	.WORD	26.	; SLEN = 32 BYTES
1990	015212	002436	.WORD	TBUF	; SEGB = TBUF
1991	015214	101400	.WORD	101400	; OWN = LUA;STP;ENP
1992	015216	000000	.WORD	0	
1993			:		
1994	015220	000032	.WORD	26.	; SLEN = 32 BYTES
1995	015222	002436	.WORD	TBUF	; SEGB = TBUF

1996	015224	101400	.WORD	101400	; OWN = LUA;STP;ENP
1997	015226	000000	.WORD	0	;
1998			:		
1999	015230	000032	.WORD	26.	; SLEN = 32 BYTES
2000	015232	002436	.WORD	TBUF	; SEGB = TBUF
2001	015234	101400	.WORD	101400	; OWN = LUA;STP;ENP
2002	015236	000000	.WORD	0	;
2003			:		
2004	015240	000032	.WORD	26.	; SLEN = 32 BYTES
2005	015242	002436	.WORD	TBUF	; SEGB = TBUF
2006	015244	101400	.WORD	101400	; OWN = LUA;STP;ENP
2007	015246	000000	.WORD	0	;
2008			:		
2009	015250	000032	.WORD	26.	; SLEN = 32 BYTES
2010	015252	002436	.WORD	TBUF	; SEGB = TBUF
2011	015254	101400	.WORD	101400	; OWN = LUA;STP;ENP
2012	015256	000000	.WORD	0	;
2013			:		
2014	015260	000032	.WORD	26.	; SLEN = 32 BYTES
2015	015262	002436	.WORD	TBUF	; SEGB = TBUF
2016	015264	101400	.WORD	101400	; OWN = LUA;STP;ENP
2017	015266	000000	.WORD	0	;
2018			:		
2019	015270	000032	.WORD	26.	; SLEN = 32 BYTES
2020	015272	002436	.WORD	TBUF	; SEGB = TBUF
2021	015274	101400	.WORD	101400	; OWN = LUA;STP;ENP
2022	015276	000000	.WORD	0	;
2023			:		
2024	015300	000032	.WORD	26.	; SLEN = 32 BYTES
2025	015302	002436	.WORD	TBUF	; SEGB = LUA;STP;ENP
2026	015304	101400	.WORD	101400	; OWN = LUA;STP;ENP
2027	015306	000000	.WORD	0	;
2028			:		
2029	015310	000032	.WORD	26.	; SLEN = 32 BYTES
2030	015312	002436	.WORD	TBUF	; SEGB = TBUF
2031	015314	101400	.WORD	101400	; OWN = LUA;STP;ENP
2032	015316	000000	.WORD	0	;
2033			:		
2034	015320	000032	.WORD	26.	; SLEN = 32 BYTES
2035	015322	002436	.WORD	TBUF	; SEGB = TBUF
2036	015324	101400	.WORD	101400	; OWN = LUA;STP;ENP
2037	015326	000000	.WORD	0	;
2038			:		
2039	015330	000032	.WORD	26.	; SLEN = 32 BYTES
2040	015332	002436	.WORD	TBUF	; SEGB = TBUF
2041	015334	101400	.WORD	101400	; OWN = LUA;STP;ENP
2042	015336	000000	.WORD	0	;
2043			:		
2044	015340	000032	.WORD	26.	; SLEN = 32 BYTES
2045	015342	002436	.WORD	TBUF	; SEGB = TBUF
2046	015344	101400	.WORD	101400	; OWN = LUA;STP;ENP
2047	015346	000000	.WORD	0	;
2048			:		
2049	015350	000032	.WORD	26.	; SLEN = 32 BYTES
2050	015352	002436	.WORD	TBUF	; SEGB = TBUF
2051	015354	101400	.WORD	101400	; OWN = LUA;STP;ENP
2052	015356	000000	.WORD	0	;

2053			:			
2054	015360	000032		.WORD	26.	; SLEN = 32 BYTES
2055	015362	002436		.WORD	TBUF	; SEGB = TBUF
2056	015364	101400		.WORD	101400	; OWN = LUA;STP;ENP
2057	015366	000000		.WORD	0	;
2058			:			
2059	015370	000032		.WORD	26.	; SLEN = 32 BYTES
2060	015372	002436		.WORD	TBUF	; SEGB = TBUF
2061	015374	101400		.WORD	101400	; OWN = LUA;STP;ENP
2062	015376	000000		.WORD	0	;
2063			:			
2064	015400	000032		.WORD	26.	; SLEN = 32 BYTES
2065	015402	002436		.WORD	TBUF	; SEGB = TBUF
2066	015404	101400		.WORD	101400	; OWN = LUA;STP;ENP
2067	015406	000000		.WORD	0	;
2068			:			
2069	015410	000032		.WORD	26.	; SLEN = 32 BYTES
2070	015412	002436		.WORD	TBUF	; SEGB = TBUF
2071	015414	101400		.WORD	101400	; OWN = LUA;STP;ENP
2072	015416	000000		.WORD	0	;
2073			:			
2074	015420	000032		.WORD	26.	; SLEN = 32 BYTES
2075	015422	002436		.WORD	TBUF	; SEGB = TBUF
2076	015424	101400		.WORD	101400	; OWN = LUA;STP;ENP
2077	015426	000000		.WORD	0	;
2078			:			
2079	015430	000032		.WORD	26.	; SLEN = 32 BYTES
2080	015432	002436		.WORD	TBUF	; SEGB = TBUF
2081	015434	101400		.WORD	101400	; OWN = LUA;STP;ENP
2082	015436	000000		.WORD	0	;
2083			:			
2084	015440	000032		.WORD	26.	; SLEN = 32 BYTES
2085	015442	002436		.WORD	TBUF	; SEGB = TBUF
2086	015444	101400		.WORD	101400	; OWN = LUA;STP;ENP
2087	015446	000000		.WORD	0	;
2088			:			
2089	015450	000032		.WORD	26.	; SLEN = 32 BYTES
2090	015452	002436		.WORD	TBUF	; SEGB = TBUF
2091	015454	101400		.WORD	101400	; OWN = LUA;STP;ENP
2092	015456	000000		.WORD	0	;
2093			:			
2094	015460	000032		.WORD	26.	; SLEN = 32 BYTES
2095	015462	002436		.WORD	TBUF	; SEGB = TBUF
2096	015464	101400		.WORD	101400	; OWN = LUA;STP;ENP
2097	015466	000000		.WORD	0	;
2098			:			
2099	015470	000032		.WORD	26.	; SLEN = 32 BYTES
2100	015472	002436		.WORD	TBUF	;
2101	015474	101400		.WORD	101400	;
2102	015476	000000		.WORD	0	;
2103			:			
2104	015500	000032		.WORD	26.	;
2105	015502	002436		.WORD	TBUF	;
2106	015504	101400		.WORD	101400	;
2107	015506	000000		.WORD	0	;
2108			:			
2109	015510	000032		.WORD	26.	;

```

2110 015512 002436' .WORD TBUF ;
2111 015514 101400 .WORD 101400 ;
2112 015516 000000 .WORD 0 ;
2113 ;
2114 015520 000032 .WORD 26. ;
2115 015522 002436' .WORD TBUF ;
2116 015524 101400 .WORD 101400 ;
2117 015526 000000 .WORD 0 ;
2118 ;
2119 015530 000032 .WORD 26. ;
2120 015532 002436' .WORD TBUF ;
2121 015534 101400 .WORD 101400 ;
2122 015536 000000 .WORD 0 ;
2123 ;
2124 015540 000032 .WORD 26. ;
2125 015542 002436' .WORD TBUF ;
2126 015544 101400 .WORD 101400 ;
2127 015546 000000 .WORD 0 ;
2128 ;
2129 015550 000032 .WORD 26. ; SLEN = 32 BYTES
2130 015552 002436' .WORD TBUF ; SEGB = TBUF
2131 015554 101400 .WORD 101400 ; OI'N = LUA;STP;ENP
2132 015556 000000 .WORD 0 ;
2133 ;
2134 015560 000032 .WORD 26. ; SLEN = 32 BYTES
2135 015562 002436' .WORD TBUF ; SEGB = TBUF
2136 015564 001400 .WORD 001400 ; OWN = PORT DRIVER;STP;ENP
2137 015566 000000 .WORD 0 ;
2138 ;
2139 015570 000016 ; TDRB4A: .WORD 14. ; SLEN = 14 BYTES
2140 015572 002436' .WORD TBUF ; SEGB = TBUF
2141 015574 101000 .WORD 101000 ; OWN = LUA;STP
2142 015576 000000 .WORD 0 ;
2143 ;
2144 015600 000016 .WORD 14. ; SLEN = 14 BYTES
2145 015602 002436' .WORD TBUF ; SEGB = TBUF
2146 015604 100400 .WORD 100400 ; OWN = LUA;ENP
2147 015606 000000 .WORD 0 ;
2148 ;
2149 015610 000016 .WORD 14. ; SLEN = 14 BYTES
2150 015612 002436' .WORD TBUF ; SEGB = TBUF
2151 015614 001400 .WORD 001400 ; OWN = PORT DRIVER;STP;ENP
2152 015616 000000 .WORD 0 ;
2153 ;
2154 ;:DEFAULT DATA FOR TEST11
2155 ;
2156 ;
2157 015620 000000 CRCH: .WORD 0 ; CRC STORAGE
2158 ;
2159 ;:DEFAULT UDBB FOR TEST11
2160 ;
2161 015622 002000 UDB10A: .WORD 2000 ; FLEN = 1024 WORDS
2162 015624 006440' .WORD RBUF ; HDBB = RBUF
2163 015626 000000 .WORD 0
2164 015630 000000 .WORD 0
2165 015632 000010 .WORD 10
2166
    
```

2167  
 2168  
 2169 015634  
 2170 015634 000000  
 2171 015636 002000  
 2172 015640 004000  
 2173 015642 006000  
 2174 015644 010000  
 2175 015646 012000  
 2176 015650 014000  
 2177 015652 016000  
 2178 015654 020000  
 2179 015656 022000  
 2180 015660 024000  
 2181 015662 026000  
 2182 015664 030000  
 2183 015666 032000  
 2184 015670 034000  
 2185 015672 036000  
 2186  
 2187  
 2188  
 2189  
 2190 015674 004000  
 2191 015676 000000  
 2192 015700 000000  
 2193 015702 000000  
 2194 015704 000000  
 2195  
 2196  
 2197  
 2198 015706 014000  
 2199  
 2200  
 2201  
 2202 015710 000146  
 2203  
 2204 015712 062000  
 2205 015714 066000  
 2206 015716 072000  
 2207 015720 076000  
 2208 015722 102000  
 2209 015724 106000  
 2210 015726 112000  
 2211 015730 116000  
 2212 015732 122000  
 2213 015734 126000  
 2214 015736 132000  
 2215 015740 136000  
 2216 015742 142000  
 2217 015744 146000  
 2218 015746 152000  
 2219 015750 156000  
 2220 015752 162000  
 2221 015754 166000  
 2222 015756 172000  
 2223 015760 176000

;ROM ADDRESS TABLE FOR TEST11

MEM10A: .WORD 0 ; ADDRESS OF ROM 1ST 1K  
 .WORD 2000 ; SECOND 1K  
 .WORD 4000 ; ETC.  
 .WORD 6000  
 .WORD 10000  
 .WORD 12000  
 .WORD 14000  
 .WORD 16000  
 .WORD 20000  
 .WORD 22000  
 .WORD 24000  
 .WORD 26000  
 .WORD 30000  
 .WORD 32000  
 .WORD 34000  
 .WORD 36000

;DEFAULT UDBB FOR TST12

UDB11A: .WORD 4000 ; FLEN = 1024. WORDS  
 .WORD 0 ; HDBB = RBUF OR TBUF (LOADED BY TEST)  
 .WORD 0  
 .WORD 0 ; IDBB (LOADED BY TEST)  
 .WORD 0 ; IDBB (Upper addr bits) loaded by test

;WCS DOWNLINE LOAD ADDRESS TABLE FOR TEST12

MEM11A: .WORD 14000 ; TOP 1K SECTION OF MEMORY

;INTERNAL RAM MEMORY ADDRESS TABLE FOR TEST12

.WORD END13A-MEM13A ; WORD SIZE OF MEM13A  
 MEM13A: .WORD 062000 ; FIRST 1K BLOCK OF INTERNAL RAM MEMORY  
 .WORD 066000  
 .WORD 072000  
 .WORD 076000  
 .WORD 102000  
 .WORD 106000  
 .WORD 112000  
 .WORD 116000  
 .WORD 122000  
 .WORD 126000  
 .WORD 132000  
 .WORD 136000  
 .WORD 142000  
 .WORD 146000  
 .WORD 152000  
 .WORD 156000  
 .WORD 162000  
 .WORD 166000  
 .WORD 172000  
 .WORD 176000

2224	015762	002000	.WORD	002000	;FROM HERE ON EXT ADDR BIT WILL BE
2225					;SET IN UDBB+4
2226	015764	006000	.WORD	006000	
2227	015766	012000	.WORD	012000	
2228	015770	016000	.WORD	016000	
2229	015772	022000	.WORD	022000	
2230	015774	026000	.WORD	026000	
2231	015776	032000	.WORD	032000	
2232	016000	036000	.WORD	036000	
2233	016002	042000	.WORD	042000	
2234	016004	046000	.WORD	046000	
2235	016006	052000	.WORD	052000	
2236	016010	056000	.WORD	056000	
2237	016012	062000	.WORD	062000	
2238	016014	066000	.WORD	066000	
2239	016016	072000	.WORD	072000	
2240	016020	076000	.WORD	076000	
2241	016022	102000	.WORD	102000	
2242	016024	106000	.WORD	106000	
2243	016026	111200	.WORD	111200	
2244	016030	116000	.WORD	116000	
2245	016032	122000	.WORD	122000	
2246	016034	126000	.WORD	126000	
2247	016036	132000	.WORD	132000	
2248	016040	136000	.WORD	136000	
2249	016042	142000	.WORD	142000	
2250	016044	146000	.WORD	146000	
2251	016046	152000	.WORD	152000	
2252	016050	156000	.WORD	156000	
2253	016052	162000	.WORD	162000	
2254	016054	166000	.WORD	166000	
2255	016056	172000	.WORD	172000	
2256		016060			

END13A = .

;PHYSICAL ADDRESSES FOR TEST 20

2261	016060	125252	ADR21:	.WORD	125252	; DEFAULT PHYSICAL ADDRESS
2262	016062	125252		.WORD	125252	
2263	016064	125252		.WORD	125252	

2265	016066	052524	ADR21C:	.WORD	52524	; COMPLEMENTED PHYSICAL ADDRESS
2266	016070	052525		.WORD	52525	
2267	016072	052525		.WORD	52525	

;MULTICAST ADDRESS LIST FOR TEST 21

2271	016074	125253	MULTL:	.WORD	125253	; MULTICAST ADDRESS LIST
2272	016076	125252		.WORD	125252	
2273	016100	125252		.WORD	125252	
2274	016102	125253		.WORD	125253	
2275	016104	052525		.WORD	052525	
2276	016106	125252		.WORD	125252	
2277	016110	125253		.WORD	125253	
2278	016112	125252		.WORD	125252	
2279	016114	052525		.WORD	052525	
2280	016116	125253		.WORD	125253	



2281	016120	177777	.WORD	177777
2282	016122	052525	.WORD	052525
2283	016124	125253	.WORD	125253
2284	016126	000000	.WORD	000000
2285	016130	125252	.WORD	125252
2286	016132	177777	.WORD	177777
2287	016134	000000	.WORD	000000
2288	016136	177777	.WORD	177777
2289	016140	177777	.WORD	177777
2290	016142	052525	.WORD	052525
2291	016144	125252	.WORD	125252
2292	016146	177777	.WORD	177777
2293	016150	125252	.WORD	125252
2294	016152	052525	.WORD	052525
2295	016154	177777	.WORD	177777
2296	016156	000000	.WORD	000000
2297	016160	052525	.WORD	052525
2298	016162	177777	.WORD	177777
2299	016164	177777	.WORD	177777
2300	016166	125252	.WORD	125252
2301				
2302	016170	052525	.WORD	052525
2303	016172	052525	.WORD	052525
2304	016174	052525	.WORD	052525
2305	016176	052525	.WORD	052525
2306	016200	125252	.WORD	125252
2307	016202	052525	.WORD	052525
2308	016204	052525	.WORD	052525
2309	016206	052525	.WORD	052525
2310	016210	125252	.WORD	125252
2311	016212	052525	.WORD	052525
2312	016214	000000	.WORD	000000
2313	016216	125252	.WORD	125252
2314	016220	052525	.WORD	052525
2315	016222	177777	.WORD	177777
2316	016224	052525	.WORD	052525
2317	016226	000001	.WORD	000001
2318	016230	000000	.WORD	000000
2319	016232	000000	.WORD	000000
2320	016234	000001	.WORD	000001
2321	016236	125252	.WORD	125252
2322	016240	052525	.WORD	052525
2323	016242	000001	.WORD	000001
2324	016244	052525	.WORD	052525
2325	016246	125252	.WORD	125252
2326	016250	000001	.WORD	000001
2327	016252	177777	.WORD	177777
2328	016254	125252	.WORD	125252
2329	016256	000001	.WORD	000001
2330	016260	000000	.WORD	000000
2331	016262	052525	.WORD	052525
2332				
2333				
2334				
2335	016264	000032	.WORD	26.
2336	016266	002436	.WORD	TBUF
2337	016270	021400	.WORD	021400

MULTLC:

; COMPLIMENTED ADDRESS LIST

;DEFAULT EXPECTED DATA

TDR14A:

; EXPECTED TORB FOR  
 ; TEST13,17,20-23,25,26  
 ; MTCH,STP,ENP

2338	016272	000000		.WORD	0	;
2339	016274	000040	TDR15A:	.WORD	32.	; EXPECTED TDRB FOR
2340	016276	002436'		.WORD	TBUF	; TESTS 14,15
2341	016300	021400		.WORD	021400	; MTCH,STP,ENP
2342	016302	000000		.WORD	0	
2343	016304	000016	TDR18A:	.WORD	14.	; FIRST TDRB FOR TEST18
2344	016306	002436'		.WORD	TBUF	;
2345	016310	041400		.WORD	041400	; ERR,STP,ENP
2346	016312	100000		.WORD	100000	; BUFL ERROR
2347	016314	000022	TDR18B:	.WORD	18.	; SECOND TDRB FOR TEST18
2348	016316	002436'		.WORD	TBUF	;
2349	016320	041400		.WORD	041400	; ERR,STP,ENP
2350	016322	100000		.WORD	100000	; BUFL ERROR
2351	016324	000016	TDR20A:	.WORD	14.	; FIRST TDRB FOR TEST19
2352	016326	002436'		.WORD	TBUF	;
2353	016330	001000		.WORD	001000	; STP
2354	016332	000000		.WORD	0	
2355	016334	000016	TDR20B:	.WORD	14.	; SECOND TDRB FOR TEST19
2356	016336	002436'		.WORD	TBUF	;
2357	016340	020400		.WORD	20400	; MTCH,ENP
2358	016342	000000		.WORD	0	
2359	016344	000032	TDR21X:	.WORD	26.	; EXPECTED TDRB FOR
2360	016346	002436'		.WORD	TBUF	; TESTS 20,21
2361	016350	001400		.WORD	001400	; STP,ENP
2362	016352	000000		.WORD	0	
2363	016354	000042	TDR24A:	.WORD	34.	; EXPECTED TDRB FOR
2364	016356	002436'		.WORD	TBUF	; TEST 24, 1ST PASS
2365	016360	041400		.WORD	041400	; BUFL,STP,ENP
2366	016362	100000		.WORD	100000	;
2367	016364	000050	TDR24B:	.WORD	40.	; EXPECTED TDRB FOR
2368	016366	002436'		.WORD	TBUF	; TEST 24, 2ND PASS
2369	016370	041400		.WORD	041400	; BUFL,STP,ENP
2370	016372	100000		.WORD	100000	;
2371						
2372	016374	000040	RDR14B:	.WORD	32.	; EXPECTED RDRB
2373	016376	006440'		.WORD	RBUF	; FOR TEST 14
2374	016400	001400		.WORD	001400	; STP,ENP
2375	016402	000040		.WORD	32.	;
2376	016404	000040	RDR15A:	.WORD	32.	; EXPECTED RDRB F R
2377	016406	006440'		.WORD	RBUF	; TESTS 15
2378	016410	065400		.WORD	065400	; ERRS,CRC,FRM,STP,ENP
2379	016412	000040		.WORD	32.	;
2380	016414	000040	RDR17A:	.WORD	32.	; FIRST RDRB FOR TEST17
2381	016416	006440'		.WORD	RBUF	;
2382	016420	001400		.WORD	001400	; ERRS,STP,ENP
2383	016422	020036		.WORD	020036	; NCHN
2384	016424	000040	RDR17B:	.WORD	32.	; SECOND RDRB FOR TEST17
2385	016426	006440'		.WORD	RBUF	;
2386	016430	100000		.WORD	100000	; OWN = DELUA
2387	016432	000000		.WORD	0	
2388	016434	000020	RDR20A:	.WORD	16.	; FIRST RDRB FOR TEST19
2389	016436	006440'		.WORD	RBUF	;
2390	016440	001000		.WORD	001000	; STP
2391	016442	000000		.WORD	0	
2392	016444	000026	RDR20B:	.WORD	22.	; SECOND RDRB FOR TEST19
2393	016446	007040'		.WORD	RBUF2	;
2394	016450	000400		.WORD	000400	; ENP

```

2395 016452 000040          .WORD 32.
2396 016454 000040          .WORD 32. ; TEST13,20-23,25-26
2397 016456 006440          .WORD RBUF
2398 016460 001400          .WORD 1400 ; STP, ENP
2399 016462 000036          .WORD 30.
2400
2401
2402
2403 016464 100114          MODE15: .WORD 100114 ; MODE = PROM,DTCR,INTL
2404 016466 120104          MODE17: .WORD 120104 ; MODE = PROM,DRDC,INTL
2405 016470 100000          MODE20: .WORD 100000 ; MODE = PROM
2406 016472 000104          MODE21: .WORD 104 ; INTL LOOPBACK ONLY
2407 016474 040104          MODE24: .WORD 040104 ; MODE = ENAL,INTL
2408 016476 110104          MODE25: .WORD 110104 ; MODE = PROM,TPAD,INTL
2409 016500 000002          UDB28A: .WORD 2 ; UDBB FOR TEST26
2410 016502 000000          .WORD 0 ;
2411 016504 000000          .WORD 0 ;
2412 016506 000000          .WORD 0 ;
2413 016510 000000          .WORD 0 ;
2414 016512 021040          SWADDR: .WORD 21040 ; SWITCH PACK ADDRESS
2415          ;GLOBAL DATA AND FLAGS
2416          ;
2417 016514 000000          EPCSR0: .WORD 0 ; PCSRO AT TIME OF ERROR
2418 016516 000000          EPCSR1: .WORD 0 ; PCSR1 AT TIME OF ERROR
2419 016520 000000          ERDRB0: .WORD 0 ; RDRB+0 AT TIME OF ERROR
2420 016522 000000          ERDRB2: .WORD 0 ; RDRB+2 AT TIME OF ERROR
2421 016524 000000          ERDRB4: .WORD 0 ; RDRB+4 AT TIME OF ERROR
2422 016526 000000          ERDRB6: .WORD 0 ; RDRB+6 AT TIME OF ERROR
2423 016530 000000          XRDRB0: .WORD 0 ; EXPECTED RDRB+0 AT TIME OF ERROR
2424 016532 000000          XRDRB2: .WORD 0 ; EXPECTED RDRB+2 AT TIME OF ERROR
2425 016534 000000          XRDRB4: .WORD 0 ; EXPECTED RDRB+4 AT TIME OF ERROR
2426 016536 000000          XRDRB6: .WORD 0 ; EXPECTED RDRB+6 AT TIME OF ERROR
2427 016540 000000          ETDRB0: .WORD 0 ; TDRB+0 AT TIME OF ERROR
2428 016542 000000          ETDRB2: .WORD 0 ; TDRB+2 AT TIME OF ERROR
2429 016544 000000          ETDRB4: .WORD 0 ; TDRB+4 AT TIME OF ERROR
2430 016546 000000          ETDRB6: .WORD 0 ; TDRB+6 AT TIME OF ERROR
2431 016550 000000          XTDRB0: .WORD 0 ; EXPECTED TDRB+0 AT TIME OF ERROR
2432 016552 000000          XTDRB2: .WORD 0 ; EXPECTED TDRB+2 AT TIME OF ERROR
2433 016554 000000          XTDRB4: .WORD 0 ; EXPECTED TDRB+4 AT TIME OF ERROR
2434 016556 000000          XTDRB6: .WORD 0 ; EXPECTED TDRB+6 AT TIME OF ERROR
2435
2436 016560 000000          BYTCNT: .WORD 0 ; NUMBER OF BYTES/PACKET
2437 016562 000000          DOCRC: .WORD 0 ; CRC REQUIREMENT FOR SUBROUTINES
2438          ;          0 = NO CRC
2439          ;          1 = APPEND CRC
2440
2441 016564 000000          EDAT: .WORD 0 ; ACTUAL DATA AT TIME OF ERROR
2442 016566 000000          XDAT: .WORD 0 ; EXPECTED DATA AT TIME OF ERROR
2443 016570 000000          ECRC: .WORD 0 ; ACTUAL CRC VALUE AT TIME OF ERROR
2444 016572 000000          ECRCB: .WORD 0
2445 016574 000000          XCRC: .WORD 0 ; EXPECTED CRC VALUE AT TIME OF ERROR
2446 016576 000000          XCRCB: .WORD 0
2447          ;
2448 016600 000000          ECODE: .WORD 0 ; SELF TEST ERROR CODE SHIFTED RIGHT
2449          ;
2450 016602 000000          METER: .WORD 0 ; CLOCK TICKS
2451 016604 000000          NEXMEM: .WORD 0 ; NXM TIMEOUT FLAG
    
```

2452	016606	000000	EAFLAG:	.WORD	0	; EXT ADDRESS BITS FLAG
2453	016610	000000	DNIFLG:	.WORD	0	; DNI INTERRUPT FLAG
2454	016612	000000	FRSTIM:	.WORD	0	; FIRST TIME FLAG
2455	016614	166670	POLYH:	.WORD	166670	; HIGH WORD OF PACKET CRC
2456	016616	101440	POLYL:	.WORD	101440	; LOW WORD OF PACKET CRC
2457	016620	000000	PRNTIT:	.WORD	0	; PRINT ENABLED FLAG
2458	016622		REPLY:	.BLKW	2	; DEFAULT STORAGE FOR REPLY TO
2459						; MANUAL INTFRVENTION REQUEST
2460						
2461	016626	177777	PATRN1:	.WORD	177777	; SA0_SA1 TEST PATTERN
2462	016630	000000		.WORD	0	
2463	016632	052525		.WORD	52525	
2464	016634	125252		.WORD	125252	
2465	016636	155463		.WORD	155463	
2466	016640	036334		.WORD	36334	
2467	016642	141616		.WORD	141616	
2468	016644	052525		.WORD	52525	
2469	016646	125252		.WORD	125252	
2470						
2471	016650		ERRTBL			
	016650					L\$ERRTBL::
	016650	000000	ERRTYP::	.WORD	0	
	016652	000000	ERRNBR::	.WORD	0	
	016654	000000	ERRMSG::	.WORD	0	
	016656	000000	ERRBLK::	.WORD	0	

```

2473 .SBTTL GLOBAL TEXT SECTION
2474
2475 ;**
2476 ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
2477 ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
2478 ; MORE THAN ONE TEST.
2479 ; -
2480
2481 ;
2482 ; NAMES OF DEVICES SUPPORTED BY PROGRAM
2483 ;
2484 016660 DEVTYP <DELUA>
      016660 L#DVTYP::
      016660 104 105 114 .ASCIZ «DELUA»
      016663 125 101 000 .EVEN

2485
2486 ; TEST DESCRIPTION
2487 ;
2488 ;
2489 016666 DESCRIPT <DELUA - PDP11 FUNCTIONAL DIAGNOSTIC dtd 08 JUL-85>
      016666 L#DESC::
      016666 104 105 114 .ASCIZ /DELUA - PDP11 FUNCT

IONAL DIAGNOSTIC
dtd 08-JUL-85/
      016671 125 101 040
      016674 055 040 120
      016677 104 120 061
      016702 061 040 106
      016705 125 116 103
      016710 124 111 117
      016713 116 101 114
      016716 040 104 111
      016721 101 107 116
      016724 117 123 124
      016727 111 103 040
      016732 040 144 164
      016735 144 040 060
      016740 070 055 112
      016743 125 114 055
      016746 070 065 000

      .EVEN

2490
2491
    
```

```

2493
2494          ; FORMAT STATEMENTS USED IN PRINT CALLS
2495          ;
2496
2497          ;
2498 016752      045      116      045  FRM001: .ASCIZ  /%N%APCSR%D1%A DOES NOT EXIST/
      016755      101      120      103
      016760      123      122      045
      016763      104      061      045
      016766      101      040      104
      016771      117      105      123
      016774      040      116      117
      016777      124      040      105
      017002      130      111      123
      017005      124      000
2499 017007      045      116      045  FRM002: .ASCIZ  /%N%A EXPECTED DATA = %06%N%A ACTUAL DATA = %06/
      017012      101      040      105
      017015      130      120      105
      017020      103      124      105
      017023      104      040      104
      017026      101      124      101
      017031      040      075      040
      017034      045      117      066
      017037      045      116      045
      017042      101      040      101
      017045      103      124      125
      017050      101      114      040
      017053      104      101      124
      017056      101      040      075
      017061      040      040      040
      017064      045      117      066
      017067      000
2500 017070      045      116      045  FRM003: .ASCIZ  /%N%A PCSRO = %06%N%A PCSR1 = %06/
      017073      101      040      120
      017076      103      123      122
      017101      060      040      075
      017104      040      045      117
      017107      066      045      116
      017112      045      101      040
      017115      120      103      123
      017120      122      061      040
      017123      075      040      045
      017126      117      066      000
2501 017131      045      116      045  FRM004: .ASCIZ  /%N%A SELF TEST ERROR CODE = %02/
      017134      101      040      123
      017137      105      114      106
      017142      040      124      105
      017145      123      124      040
      017150      105      122      122
      017153      117      122      040
      017156      103      117      104
      017161      105      040      075
      017164      040      045      117
      017167      062      000
2502 017171      045      116      045  FRM005: .ASCIZ  /%N%A EXPECTED TDRB.0 = %06%N%A ACTUAL TDRB.0 = %06/
      017174      101      040      105
      017177      130      120      105
    
```

	017202	103	124	105	
	017205	104	040	124	
	017210	104	122	102	
	017213	053	060	040	
	017216	075	040	045	
	017221	117	066	045	
	017224	116	045	101	
	017227	040	101	103	
	017232	124	125	101	
	017235	114	040	124	
	017240	104	122	102	
	017243	053	060	040	
	017246	075	040	040	
	017251	040	045	117	
	017254	066	000		
2503	017256	045	116	045	FRM006: .ASCIZ /%N% A EXPECTED TDRB+2 = %06%N% A ACTUAL TDRB+2 = %06/
	017261	101	040	105	
	017264	130	120	105	
	017267	103	124	105	
	017272	104	040	124	
	017275	104	122	102	
	017300	053	062	040	
	017303	075	040	045	
	017306	117	066	045	
	017311	116	045	101	
	017314	040	101	103	
	017317	124	125	101	
	017322	114	040	124	
	017325	104	122	102	
	017330	053	062	040	
	017333	075	040	040	
	017336	040	045	117	
	017341	066	000		
2504	017343	045	116	045	FRM007: .ASCIZ /%N% A EXPECTED TDRB+4 = %06%N% A ACTUAL TDRB+4 = %06/
	017346	101	040	105	
	017351	130	120	105	
	017354	103	124	105	
	017357	104	040	124	
	017362	104	122	102	
	017365	053	064	040	
	017370	075	040	045	
	017373	117	066	045	
	017376	116	045	101	
	017401	040	101	103	
	017404	124	125	101	
	017407	114	040	124	
	017412	104	122	102	
	017415	053	064	040	
	017420	075	040	040	
	017423	040	045	117	
	017426	066	000		
2505	017430	045	116	045	FRM008: .ASCIZ /%N% A EXPECTED TDRB+6 = %06%N% A ACTUAL TDRB+6 = %06/
	017433	101	040	105	
	017436	130	120	105	
	017441	103	124	105	
	017444	104	040	124	
	017447	104	122	102	

	017452	053	066	040	
	017455	075	040	045	
	017460	117	066	045	
	017463	116	045	101	
	017466	040	101	103	
	017471	124	125	101	
	017474	114	040	124	
	017477	104	122	102	
	017502	053	066	040	
	017505	075	040	040	
	017510	040	045	117	
	017513	066	000		
2506	017515	045	116	045	FRM009: .ASCIZ /%N%A EXPECTED RDRB+0 = %06%N%A ACTUAL RDRB+0 = %06/
	017520	101	040	105	
	017523	130	120	105	
	017526	103	124	105	
	017531	104	040	122	
	017534	104	122	102	
	017537	053	060	040	
	017542	075	040	045	
	017545	117	066	045	
	017550	116	045	101	
	017553	040	101	103	
	017556	124	125	101	
	017561	114	040	122	
	017564	104	122	102	
	017567	053	060	040	
	017572	075	040	040	
	017575	040	045	117	
	017600	066	000		
2507	017602	045	116	045	FRM010: .ASCIZ /%N%A EXPECTED RDRB+2 = %06%N%A ACTUAL RDRB+2 = %06/
	017605	101	040	105	
	017610	130	120	105	
	017613	103	124	105	
	017616	104	040	122	
	017621	104	122	102	
	017624	053	062	040	
	017627	075	040	045	
	017632	117	066	045	
	017635	116	045	101	
	017640	040	101	103	
	017643	124	125	101	
	017646	114	040	122	
	017651	104	122	102	
	017654	053	062	040	
	017657	075	040	040	
	017662	040	045	117	
	017665	066	000		
2508	017667	045	116	045	FRM011: .ASCIZ /%N%A EXPECTED RDRB+4 = %06%N%A ACTUAL RDRB+4 = %06/
	017672	101	040	105	
	017675	130	120	105	
	017700	103	124	105	
	017703	104	040	122	
	017706	104	122	102	
	017711	053	064	040	
	017714	075	040	045	
	017717	117	066	045	



	017722	116	045	101	
	017725	040	101	103	
	017730	124	125	101	
	017733	114	040	122	
	017736	104	122	102	
	017741	053	064	040	
	017744	075	040	040	
	017747	040	045	117	
	017752	066	000		
2509	017754	045	116	045	FRM012: .ASCIZ /NNA EXPECTED RDRB+6 = 06NNA ACTUAL RDRB+6 = 06/
	017757	101	040	105	
	017762	130	120	105	
	017765	103	124	105	
	017770	104	040	122	
	017773	104	122	102	
	017776	053	066	040	
	020001	075	040	045	
	020004	117	066	045	
	020007	116	045	101	
	020012	040	101	103	
	020015	124	125	101	
	020020	114	040	122	
	020023	104	122	102	
	020026	053	066	040	
	020031	075	040	040	
	020034	040	045	117	
	020037	066	000		
2510	020041	045	116	045	FRM013: .ASCIZ /NNA EXPECTED CRC = 06NNA 06/
	020044	101	040	105	
	020047	130	120	105	
	020052	103	124	105	
	020055	104	040	103	
	020060	122	103	040	
	020063	075	040	045	
	020066	117	066	045	
	020071	116	045	101	
	020074	040	040	040	
	020077	040	040	040	
	020102	040	040	040	
	020105	040	040	040	
	020110	040	040	040	
	020113	040	045	117	
	020116	066	000		
2511	020120	045	116	045	FRM014: .ASCIZ /NNA ACTUAL CRC = 06NNA 06/
	020123	101	040	101	
	020126	103	124	125	
	020131	101	114	040	
	020134	103	122	103	
	020137	040	040	040	
	020142	075	040	045	
	020145	117	066	045	
	020150	116	045	101	
	020153	040	040	040	
	020156	040	040	040	
	020161	040	040	040	
	020164	040	040	040	
	020167	040	040	040	

	020172	040	045	117		
	020175	066	060			
2512	020177	045	116	045	FRM015: .ASCIZ	/*N*T/
	020202	124	000			
2513	020204	045	116	045	FRM016: .ASCIZ	/*N*AROM MICROCODE VERSION (DECIMAL): *D2/
	020207	101	122	117		
	020212	115	040	115		
	020215	111	103	122		
	020220	117	103	117		
	020223	104	105	040		
	020226	126	105	122		
	020231	123	111	117		
	020234	116	040	050		
	020237	104	105	103		
	020242	111	115	101		
	020245	114	051	072		
	020250	040	045	104		
	020253	062	000			
2514	020255	045	116	045	FRM017: .ASCIZ	/*N*ASWITCH PACK = *06/
	020260	101	123	127		
	020263	111	124	103		
	020266	110	040	120		
	020271	101	103	113		
	020274	040	075	040		
	020277	045	117	066		
	020302	000				
2515	020303	045	116	045	FRM018: .ASCIZ	/*N*APORT STATUS WORD 1: *06/
	020306	101	120	117		
	020311	122	124	040		
	020314	123	124	101		
	020317	124	125	123		
	020322	040	127	117		
	020325	122	104	040		
	020330	061	072	040		
	020333	045	117	066		
	020336	000				
2516	020337	045	116	045	FRM019: .ASCIZ	/*N*A WORD 2: *06/
	020342	101	040	040		
	020345	040	040	040		
	020350	040	040	040		
	020353	040	040	040		
	020356	040	127	117		
	020361	122	104	040		
	020364	062	072	040		
	020367	045	117	066		
	020372	000				
2517	020373	045	116	045	FRM020: .ASCIZ	/*N*A WORD 3: *06/
	020376	101	040	040		
	020401	040	040	040		
	020404	040	040	040		
	020407	040	040	040		
	020412	040	127	117		
	020415	122	104	040		
	020420	063	072	040		
	020423	045	117	066		
	020426	000				
2518	020427	045	116	045	FRM021: .ASCIZ	/*N*A WORD 4: *06/

	020432	101	040	040	
	020435	040	040	040	
	020440	040	040	040	
	020443	040	040	040	
	020446	040	127	117	
	020451	122	104	040	
	020454	064	072	040	
	020457	045	117	066	
	020462	000			
2519	020463	045	116	045	FRM022: .ASCIZ /%N% A EXPECTED UDBB+4 = > 0 %N% A ACTUAL UDBB+4 = %06/
	020466	101	040	105	
	020471	130	120	105	
	020474	103	124	105	
	020477	104	040	125	
	020502	104	102	102	
	020505	053	064	040	
	020510	075	040	076	
	020513	040	060	040	
	020516	045	116	045	
	020521	101	040	101	
	020524	103	124	125	
	020527	101	114	040	
	020532	125	104	102	
	020535	102	053	064	
	020540	040	075	040	
	020543	045	117	066	
	020546	000			
2520	020547	045	116	045	FRM023: .ASCIZ /%N% A PCSRO = %06/
	020552	101	040	120	
	020555	103	123	122	
	020560	060	040	075	
	020563	040	045	117	
	020566	066	000		
2521					
2522	020570	045	116	045	MSG1: .ASCII /%N% A FAILURE TO INSTALL EXT. LOOPBACK CONN. MAY RESULT IN/
	020573	101	106	101	
	020576	111	114	125	
	020601	122	105	040	
	020604	124	117	040	
	020607	111	116	123	
	020612	124	101	114	
	020615	114	040	105	
	020620	130	124	056	
	020623	040	114	117	
	020626	117	120	102	
	020631	101	103	113	
	020634	040	103	117	
	020637	116	116	056	
	020642	040	115	101	
	020645	131	040	122	
	020650	105	123	125	
	020653	114	124	040	
	020656	111	116		
2523	020660	045	116	045	.ASCIZ /%N% ANETWORK FAULTS%N/
	020663	101	116	105	
	020666	124	127	117	
	020671	122	113	040	

	020674	106	101	125	
	020677	114	124	123	
	020702	045	116	000	
2524	020705	111	116	123	MNMSG1: .ASCIZ /INSTALL LOOPBACK CONNECTOR, THEN PRESS <CR> TO CONTINUE/
	020710	124	101	114	
	020713	114	040	114	
	020716	117	117	120	
	020721	102	101	103	
	020724	113	040	103	
	020727	117	116	116	
	020732	105	103	124	
	020735	117	122	054	
	020740	040	124	110	
	020743	105	116	040	
	020746	120	122	105	
	020751	123	123	040	
	020754	074	103	122	
	020757	076	040	124	
	020762	117	040	103	
	020765	117	116	124	
	020770	111	116	125	
	020773	105	000		
2525	020775	105	130	124	SKIP: .ASCIZ /EXT.LOOPBACK TEST- EXT MODE NOT SELECTED - SKIP /
	021000	056	114	117	
	021003	117	120	102	
	021006	101	103	113	
	021011	040	124	105	
	021014	123	124	055	
	021017	040	105	130	
	021022	124	040	115	
	021025	117	104	105	
	021030	040	116	117	
	021033	124	040	123	
	021036	105	114	105	
	021041	103	124	105	
	021044	104	040	055	
	021047	040	123	113	
	021052	111	120	040	
	021055	000			
2526	021056	105	130	124	SKIP26: .ASCIZ /EXT.LOOPBACK TEST - MUST BE ATTENDED MODE - SKIP /
	021061	056	114	117	
	021064	117	120	102	
	021067	101	103	113	
	021072	040	124	105	
	021075	123	124	040	
	021100	055	040	115	
	021103	125	123	124	
	021106	040	102	105	
	021111	040	101	124	
	021114	124	105	116	
	021117	104	105	104	
	021122	040	115	117	
	021125	104	105	040	
	021130	055	040	123	
	021133	113	111	120	
2527	021136	040	000		

```

2528                                     .EVEN
2529
2530                                     ;*****
2531                                     ;   MESSAGES USED ONLY IN THE INITIALIZE ROUTINE.
2532                                     ;*****
2533
2534 021140      103      101      116  NOCLK:  .ASCIZ/CANNOT CONTINUE - NEED LINE CLOCK/
      021143      116      117      124
      021146      040      103      117
      021151      116      124      111
      021154      116      125      105
      021157      040      055      040
      021162      116      105      105
      021165      104      040      114
      021170      111      116      105
      021173      040      103      114
      021176      117      103      113
      021201      000
2535
2536 021202      103      101      116  M68FLD: .ASCII/CANNOT CONTINUE - NO DNI AFTER RESET/
      021205      116      117      124
      021210      040      103      117
      021213      116      124      111
      021216      116      125      105
      021221      040      055      040
      021224      116      117      040
      021227      104      116      111
      021232      040      101      106
      021235      124      105      122
      021240      040      122      105
      021243      123      105      124
2537 021246      040      040      040      .ASCIZ/  MICROPROCESSOR SUBSYSTEM FAULT/
      021251      115      111      103
      021254      122      117      120
      021257      122      117      103
      021262      105      123      123
      021265      117      122      040
      021270      123      125      102
      021273      123      131      123
      021276      124      105      115
      021301      040      105      101
      021304      125      114      124
      021307      000
2538
2539 021310      103      101      116  DEVUNI: .ASCII/CANNOT CONTINUE - NO DNI AFTER RESET/
      021313      116      117      124
      021316      040      103      117
      021321      116      124      111
      021324      116      125      105
      021327      040      055      040
      021332      116      117      040
      021335      104      116      111
      021340      040      101      106
      021343      124      105      122
      021346      040      122      105
      021351      123      105      124
2540 021354      040      040      040      .ASCIZ/  DEVICE OR UNIBUS ERROR/
    
```

	021357	104	105	126	
	021362	111	103	105	
	021365	040	117	122	
	021370	040	125	116	
	021373	111	102	125	
	021376	123	040	105	
	021401	122	122	117	
	021404	122	000		
2541					
2542	021406	103	101	116	NIUNIB: .ASCII/CANNOT CONTINUE - NO DNI AFTER RESET/
	021411	116	117	124	
	021414	040	103	117	
	021417	116	124	111	
	021422	116	125	105	
	021425	040	055	040	
	021430	116	117	040	
	021433	104	116	111	
	021436	040	101	106	
	021441	124	105	122	
	021444	040	122	105	
	021447	123	105	124	
2543	021452	040	040	040	.ASCIZ/ NI OR UNIBUS HALTED/
	021455	116	111	040	
	021460	117	122	040	
	021463	125	116	111	
	021466	102	125	123	
	021471	040	110	101	
	021474	114	124	105	
	021477	104	000		
2544					
2545	021501	103	101	116	UNDFND: .ASCII/CANNOT CONTINUE - DNI, FATAL, AND USCI BITS/
	021504	116	117	124	
	021507	040	103	117	
	021512	116	124	111	
	021515	116	125	105	
	021520	040	055	040	
	021523	104	116	111	
	021526	054	040	106	
	021531	101	124	101	
	021534	114	054	040	
	021537	101	116	104	
	021542	040	125	123	
	021545	103	111	040	
	021550	102	111	124	
	021553	123			
2546	021554	111	116	040	.ASCIZ/IN ILLEGAL STATE/
	021557	111	114	114	
	021562	105	107	101	
	021565	114	040	123	
	021570	124	101	124	
	021573	105	000		
2547					
2548	021575	103	101	116	DNICLR: .ASCIZ/CANNOT CONTINUE - DNI WOULD NOT CLEAR FOLLOWING RESET/
	021600	116	117	124	
	021603	040	103	117	
	021606	116	124	111	
	021611	116	125	105	

021614	040	055	040
021617	104	116	111
021622	040	127	117
021625	125	114	104
021630	040	116	117
021633	124	040	103
021636	114	105	101
021641	122	040	106
021644	117	114	114
021647	117	127	111
021652	116	107	040
021655	122	105	123
021660	105	124	000

2549  
2550

.EVEN

```

2552          .SBTTL GLOBAL ERROR REPORT SECTION
2553
2554          ;++
2555          ; THE GLOBAL ERROR REPORT SECTION CONTAINS MESSAGE PRINTING AREAS
2556          ; USED BY MORE THAN TEST TO OUTPUT ADDITIONAL ERROR INFORMATION. PRINTB
2557          ; (BASIC) AND PRINTX (EXTENDED) CALLS ARE USED TO CALL PRINT SERVICES.
2558          ;--
2559
2560
2561
2562 021664      BGNMSG MSG001
2563 021664      PRINTB #FRM001,R2
2564 021664 010246
2565 021666 012746 016752'
2566 021672 012746 000002
2567 021676 010600
2568 021700 104414
2569 021702 062706 000006
2570 021706      ENDMSG
2571 021706 104423
2572
2573          ;
2574 021710      BGNMSG MSG002
2575 021710      PRINTB #FRM002,R3,R4
2576 021710 010446
2577 021712 010346
2578 021714 012746 017007'
2579 021720 012746 000003
2580 021724 010600
2581 021726 104414
2582 021730 062706 000010
2583 021734      ENDMSG
2584 021734 104423
2585
2586          ;
2587 021736      BGNMSG MSG003
2588 021736      PRINTB #FRM003,EPCSR0,EPCSR1
2589 021736 013746 016516'
2590 021742 013746 016514'
2591 021746 012746 017070'
2592 021752 012746 000003
2593 021756 010600
2594 021760 104414
2595 021762 062706 000010
2596 021766      ENDMSG
2597 021766 104423
2598
2599          ;
2600 021770      BGNMSG MSG004
2601 021770      PRINTB #FRM004,ECODE
2602 021770 013746 016600'
2603 021774 012746 017131'
2604 022000 012746 000002
    
```

```

MSG001::
    MOV R2,-(SP)
    MOV #FRM001,-(SP)
    MOV #2,-(SP)
    MOV SP,R0
    TRAP C#PNTB
    ADD #6,SP

L10002:
    TRAP C#MSG

MSG002::
    MOV R4,-(SP)
    MOV R3,-(SP)
    MOV #FRM002,-(SP)
    MOV #3,-(SP)
    MOV SP,R0
    TRAP C#PNTB
    ADD #10,SP

L10003:
    TRAP C#MSG

MSG003::
    MOV EPCSR1,-(SP)
    MOV EPCSR0,(SP)
    MOV #FRM003,(SP)
    MOV #3,-(SP)
    MOV SP,R0
    TRAP C#PNTB
    ADD #10,SP

L10004:
    TRAP C#MSG

MSG004::
    MOV ECODE,-(SP)
    MOV #FRM004,-(SP)
    MOV #2,-(SP)
    
```



	022004	010600					MOV	SP,RO
	022006	104414					TRAP	C#PNTB
	022010	062706	000006				ADD	#6,SP
2576	022014			PRINTB	#FRM015,STMSG			
	022014	013746	036556'				MOV	STMSG,-(SP)
	022020	012746	020177'				MOV	#FRM015,-(SP)
	022024	012746	000002				MOV	#2,-(SP)
	022030	010600					MOV	SP,RO
	022032	104414					TRAP	C#PNTB
	022034	062706	000006				ADD	#6,SP
2577	022040			ENDMSG				
	022040					L10005:		
	022040	104423					TRAP	C#MSG
2578								
2579	022042			BGNMSG	MSG005			
	022042					MSG005::		
2580	022042			PRINTB	#FRM005,XTDRB0,ETDRB0			
	022042	013746	016540'				MOV	ETDRB0,-(SP)
	022046	013746	016550'				MOV	XTDRB0,-(SP)
	022052	012746	017171'				MOV	#FRM005,-(SP)
	022056	012746	000003				MOV	#3,-(SP)
	022062	010600					MOV	SP,RO
	022064	104414					TRAP	C#PNTB
	022066	062706	000010				ADD	#10,SP
2581	022072			PRINTB	#FRM006,XTDRB2,ETDRB2			
	022072	013746	016542'				MOV	ETDRB2,-(SP)
	022076	013746	016552'				MOV	XTDRB2,-(SP)
	022102	012746	017256'				MOV	#FRM006,-(SP)
	022106	012746	000003				MOV	#3,-(SP)
	022112	010600					MOV	SP,RO
	022114	104414					TRAP	C#PNTB
	022116	062706	000010				ADD	#10,SP
2582	022122			PRINTB	#FRM007,XTDRB4,ETDRB4			
	022122	013746	016544'				MOV	ETDRB4,-(SP)
	022126	013746	016554'				MOV	XTDRB4,-(SP)
	022132	012746	017343'				MOV	#FRM007,-(SP)
	022136	012746	000003				MOV	#3,-(SP)
	022142	010600					MOV	SP,RO
	022144	104414					TRAP	C#PNTB
	022146	062706	000010				ADD	#10,SP
2583	022152			PRINTB	#FRM008,XTDRB6,ETDRB6			
	022152	013746	016546'				MOV	ETDRB6,-(SP)
	022156	013746	016556'				MOV	XTDRB6,-(SP)
	022162	012746	017430'				MOV	#FRM008,-(SP)
	022166	012746	000003				MOV	#3,-(SP)
	022172	010600					MOV	SP,RO
	022174	104414					TRAP	C#PNTB
	022176	062706	000010				ADD	#10,SP
2584	022202			ENDMSG				
	022202					L10006:		
	022202	104423					TRAP	C#MSG
2585								
2586	022204			BGNMSG	MSG006			
	022204					MSG006::		
2587	022204			PRINTB	#FRM009,XRDRB0,ERDRB0			
	022204	013746	016520'				MOV	ERDRB0,-(SP)
	022210	013746	016530'				MOV	XRDRB0,-(SP)

	022214	012746	017515'			MOV	#FRM009, -(SP)
	022220	012746	000003			MOV	#3, -(SP)
	022224	010600				MOV	SP, R0
	022226	104414				TRAP	C#PNTB
	022230	062706	000010			ADD	#10, SP
2588	022234			PRINTB	#FRM010, XRDRB2, ERDRB2		
	022234	013746	016522'			MOV	ERDRB2, -(SP)
	022240	013746	016532'			MOV	XRDRB2, -(SP)
	022244	012746	017602'			MOV	#FRM010, -(SP)
	022250	012746	000003			MOV	#3, -(SP)
	022254	010600				MOV	SP, R0
	022256	104414				TRAP	C#PNTB
	022260	062706	000010			ADD	#10, SP
2589	022264			PRINTB	#FRM011, XRDRB4, ERDRB4		
	022264	013746	016524'			MOV	ERDRB4, -(SP)
	022270	013746	016534'			MOV	XRDRB4, -(SP)
	022274	012746	017667'			MOV	#FRM011, -(SP)
	022300	012746	000003			MOV	#3, -(SP)
	022304	010600				MOV	SP, R0
	022306	104414				TRAP	C#PNTB
	022310	062706	000010			ADD	#10, SP
2590	022314			PRINTB	#FRM012, XRDRB6, ERDRB6		
	022314	013746	016526'			MOV	ERDRB6, -(SP)
	022320	013746	016536'			MOV	XRDRB6, -(SP)
	022324	012746	017754'			MOV	#FRM012, -(SP)
	022330	012746	000003			MOV	#3, -(SP)
	022334	010600				MOV	SP, R0
	022336	104414				TRAP	C#PNTB
	022340	062706	000010			ADD	#10, SP
2591	022344			ENDMSG			
	022344					L10007:	
	022344	104423				TRAP	C#MSG
2592							
2593	022346			BGNMSG	MSG007		
	022346					MSG007::	
2594	022346			PRINTB	#FRM002, XDAT, EDAT		
	022346	013746	016564'			MOV	EDAT, -(SP)
	022352	013746	016566'			MOV	XDAT, (SP)
	022356	012746	017007'			MOV	#FRM002, -(SP)
	022362	012746	000003			MOV	#3, -(SP)
	022366	010600				MOV	SP, R0
	022370	104414				TRAP	C#PNTB
	022372	062706	000010			ADD	#10, SP
2595	022376			ENDMSG			
	022376					L10010:	
	022376	104423				TRAP	C#MSG
2596							
2597	022400			BGNMSG	MSG008		
	022400					MSG008::	
2598	022400			PRINTB	#FRM013, XCRC, XCRCB		
	022400	013746	016576'			MOV	XCRCB, -(SP)
	022404	013746	016574'			MOV	XCRC, -(SP)
	022410	012745	020041'			MOV	#FRM013, -(SP)
	022414	012746	000003			MOV	#3, -(SP)
	022420	010600				MOV	SP, R0
	022422	104414				TRAP	C#PNTB
	022424	062706	000010			ADD	#10, SP

2599	022430			PRINTB	#FRM014,ECRC,ECRCB		
	022430	013746	016572'			MOV	ECRCB,-(SP)
	022434	013746	016570'			MOV	ECRC,-(SP)
	022440	012746	020120'			MOV	#FRM014,-(SP)
	022444	012746	000003			MOV	#3,-(SP)
	022450	010600				MOV	SP,R0
	022452	104414				TRAP	C#PNTB
	022454	062706	000010			ADD	#10,SP
2600	022460			ENDMSG		L10011:	
	022460					TRAP	C#MSG
	022460	104423					
2601							
2602	022462			BGNMSG	MSG009	MSG009::	
	022462						
2603	022462			PRINTB	#FRM018,PCBB		
	022462	013746	000300'			MOV	PCBB,-(SP)
	022466	012746	020303'			MOV	#FRM018,-(SP)
	022472	012746	000002			MOV	#2,-(SP)
	022476	010600				MOV	SP,R0
	022500	104414				TRAP	C#PNTB
	022502	062706	000006			ADD	#6,SP
2604	022506			PRINTB	#FRM019,PCBB+2		
	022506	013746	000302'			MOV	PCBB+2,-(SP)
	022512	012746	020337'			MOV	#FRM019,-(SP)
	022516	012746	000002			MOV	#2,-(SP)
	022522	010600				MOV	SP,R0
	022524	104414				TRAP	C#PNTB
	022526	062706	000006			ADD	#6,SP
2605	022532			PRINTB	#FRM020,PCBB+4		
	022532	013746	000304'			MOV	PCBB+4,-(SP)
	022536	012746	020373'			MOV	#FRM020,-(SP)
	022542	012746	000002			MOV	#2,-(SP)
	022546	010600				MOV	SP,R0
	022550	104414				TRAP	C#PNTB
	022552	062706	000006			ADD	#6,SP
2606	022556			PRINTB	#FRM021,PCBB+6		
	022556	013746	000306'			MOV	PCBB+6,-(SP)
	022562	012746	020427'			MOV	#FRM021,-(SP)
	022566	012746	000002			MOV	#2,-(SP)
	022572	010600				MOV	SP,R0
	022574	104414				TRAP	C#PNTB
	022576	062706	000006			ADD	#6,SP
2607	022602			ENDMSG		L10012:	
	022602					TRAP	C#MSG
	022602	104423					
2608							
2609	022604			BGNMSG	MSG010	MSG010::	
	022604						
2610	022604			PRINTB	#FRM022,UDBB+4		
	022604	013746	000314'			MOV	UDBB+4,(SP)
	022610	012746	020463'			MOV	#FRM022,-(SP)
	022614	012746	000002			MOV	#2,-(SP)
	022620	010600				MOV	SP,R0
	022622	104414				TRAP	C#PNTB
	022624	062706	000006			ADD	#6,SP
2611	022630			ENDMSG		L10013:	
	022630						

	022630	104423							TRAP	C\$MSG
2612										
2613	022632				BGNMSG	MSG011				
	022632								MSG011::	
2614	022632					PRINTB	#FRM023,EPCSR0			
	022632	013746	016514						MOV	EPCSR0,-(SP)
	022636	012746	020547						MOV	#FRM023,-(SP)
	022642	012746	000002						MOV	#2,-(SP)
	022646	010600							MOV	SP,R0
	022650	104414							TRAP	C\$PNTB
	022652	062706	000006						ADD	#6,SP
2615	022656				ENDMSG					
	022656								L10014:	
	022656	104423							TRAP	C\$MSG
2616										
2617					.EVEN					
2618					i					
2619	022660	015	012	122	ERR001:	.ASCIZ	<15><12>/REGISTER ACCESS ERROR/			
	022663	105	107	111						
	022666	123	124	105						
	022671	122	040	101						
	022674	103	103	105						
	022677	123	123	040						
	022702	105	122	122						
	022705	117	122	000						
2620	022710	015	012	104	ERR002:	.ASCIZ	<15><12>/DATA COMPARE ERROR IN PCSR2/			
	022713	101	124	101						
	022716	040	103	117						
	022721	115	120	101						
	022724	122	105	040						
	022727	105	122	122						
	022732	117	122	040						
	022735	111	116	040						
	022740	120	103	123						
	022743	122	062	000						
2621	022746	015	012	104	ERR003:	.ASCIZ	<15><12>/DATA COMPARE ERROR IN PCSR3/			
	022751	101	124	101						
	022754	040	103	117						
	022757	115	120	101						
	022762	122	105	040						
	022765	105	122	122						
	022770	117	122	040						
	022773	111	116	040						
	022776	120	103	123						
	023001	122	063	000						
2622	023004	015	012	123	ERR005:	.ASCIZ	<15><12>/SELF TEST FAILURE/			
	023007	105	114	106						
	023012	040	124	105						
	023015	123	124	040						
	023020	106	101	111						
	023023	114	125	122						
	023026	105	000							
2623	023030	015	012	127	ERR006:	.ASCIZ	<15><12>/WRITING ONE TO CLEAR DNI BIT FAILED/			
	023033	122	111	124						
	023036	111	116	107						
	023041	040	117	116						
	023044	105	040	124						

	023047	117	040	103	
	023052	114	105	101	
	023055	122	040	104	
	023060	116	111	040	
	023063	102	111	124	
	023066	040	106	101	
	023071	111	114	105	
	023074	104	000		
2624	023076	015	012	116	ERR007: .ASCII <15><12>/NO DNI INTERRUPT OCCURRED /
	023101	117	040	104	
	023104	116	111	040	
	023107	111	116	124	
	023112	105	122	122	
	023115	125	120	124	
	023120	040	117	103	
	023123	103	125	122	
	023126	122	105	104	
	023131	040			
2625	023132	101	106	124	.ASCIZ /AFTER GET PCBB PORT COMMAND/
	023135	105	122	040	
	023140	107	105	124	
	023143	040	120	103	
	023146	102	102	040	
	023151	120	117	122	
	023154	124	040	103	
	023157	117	115	115	
	023162	101	116	104	
	023165	000			
2626	023166	015	012	104	ERR008: .ASCII <15><12>/DNI BIT FAILED TO SET AFTER /
	023171	116	111	040	
	023174	102	111	124	
	023177	040	106	101	
	023202	111	114	105	
	023205	104	040	124	
	023210	117	040	123	
	023213	105	124	040	
	023216	101	106	124	
	023221	105	122	040	
2627	023224	116	117	120	.ASCIZ /NOP PORT COMMAND/
	023227	040	120	117	
	023232	122	124	040	
	023235	103	117	115	
	023240	115	101	116	
	023243	104	000		
2628	023245	015	012	104	ERR009: .ASCII <15><12>/DNI BIT FAILED TO SET AFTER /
	023250	116	111	040	
	023253	102	111	124	
	023256	040	106	101	
	023261	111	114	105	
	023264	104	040	124	
	023267	117	040	123	
	023272	105	124	040	
	023275	101	106	124	
	023300	105	122	040	
2629	023303	107	105	124	.ASCIZ /GET PCBB PORT COMMAND/
	023306	040	120	103	
	023311	102	102	040	

	023314	120	117	122	
	023317	124	040	103	
	023322	117	115	115	
	023325	101	116	104	
	023330	000			
2630	023331	015	012	104	ERR010: .ASCII <15><12>/DNI BIT FAILED TO SET AFTER /
	023334	116	111	040	
	023337	102	111	124	
	023342	040	106	101	
	023345	111	114	105	
	023350	104	040	124	
	023353	117	040	123	
	023356	105	124	040	
	023361	101	106	124	
	023364	105	122	040	
2631	023367	107	105	124	.ASCIZ /GET CMD PORT COMMAND/
	023372	040	103	115	
	023375	104	040	120	
	023400	117	122	124	
	023403	040	103	117	
	023406	115	115	101	
	023411	116	104	000	
2632					
2633	023414	015	012	115	ERR011: .ASCIZ <15><12>/M68000 SUBSYSTEM FAILURE/
	023417	066	070	060	
	023422	060	060	040	
	023425	123	125	102	
	023430	123	131	123	
	023433	124	105	115	
	023436	040	106	101	
	023441	111	114	125	
	023444	122	105	000	
2634					
2635	023447	015	012	104	ERR012: .ASCII <15><12>/DNI BIT FAILED TO SET AFTER /
	023452	116	111	040	
	023455	102	111	124	
	023460	040	106	101	
	023463	111	114	105	
	023466	104	040	124	
	023471	117	040	123	
	023474	105	124	040	
	023477	101	106	124	
	023502	105	122	040	
2636	023505	123	124	101	.ASCIZ /START PORT COMMAND/
	023510	122	124	040	
	023513	120	117	122	
	023516	124	040	103	
	023521	117	115	115	
	023524	101	116	104	
	023527	000			
2637	023530	015	012	124	ERR013: .ASCIZ <15><12>/TXI BIT FAILED TO SET /
	023533	130	111	040	
	023536	102	111	124	
	023541	040	106	101	
	023544	111	114	105	
	023547	104	040	124	
	023552	117	040	123	

	023555	105	124	040	
	023560	000			
2638	023561	015	012	127	ERR014: .ASCIZ <15><12>/WRITING ONE TO CLEAR TXI BI, FAILED/
	023564	122	111	124	
	023567	111	116	107	
	023572	040	117	116	
	023575	105	040	124	
	023600	117	040	103	
	023603	114	105	101	
	023606	122	040	124	
	023611	130	111	040	
	023614	102	111	124	
	023617	040	106	101	
	023622	111	114	105	
	023625	104	000		
2639	023627	015	012	122	ERR015: .ASCIZ <15><12>/RXI BIT FAILED TO SET /
	023632	130	111	040	
	023635	102	111	124	
	023640	040	106	101	
	023643	111	114	105	
	023646	104	040	124	
	023651	117	040	123	
	023654	105	124	040	
	023657	000			
2640	023660	015	012	127	ERR016: .ASCIZ <15><12>/WRITING ONE TO CLEAR RXI BIT FAILED/
	023663	122	111	124	
	023666	111	116	107	
	023671	040	117	116	
	023674	105	040	124	
	023677	117	040	103	
	023702	114	105	101	
	023705	122	040	122	
	023710	130	111	040	
	023713	102	111	124	
	023716	040	106	101	
	023721	111	114	105	
	023724	104	000		
2641	023726	015	012	124	ERR017: .ASCII <15><12>/TIMEOUT ERROR - DELUA FAILED TO /
	023731	111	115	105	
	023734	117	125	124	
	023737	040	105	122	
	023742	122	117	122	
	023745	040	055	040	
	023750	104	105	114	
	023753	125	101	040	
	023756	106	101	111	
	023761	114	105	104	
	023764	040	124	117	
	023767	040			
2642	023770	122	105	114	.ASCIZ /RELINQUISH OWNERSHIP OF RDRB /
	023773	111	116	121	
	023776	125	111	123	
	024001	110	040	117	
	024004	127	116	105	
	024007	122	123	110	
	024012	111	120	040	
	024015	117	106	040	

	024020	122	104	122	
	024023	102	040	000	
2643	024026	015	012	124	ERR018: .ASCII <15><12>/TIMEOUT ERROR - DELUA FAILED TO /
	024031	111	115	105	
	024034	117	125	124	
	024037	040	105	122	
	024042	122	117	122	
	024045	040	055	040	
	024050	104	105	114	
	024053	125	101	040	
	024056	106	101	111	
	024061	114	105	104	
	024064	040	124	117	
	024067	040			
2644	024070	122	105	114	.ASCIZ /RELINQUISH OWNERSHIP OF TDRB /
	024073	111	116	121	
	024076	125	111	123	
	024101	110	040	117	
	024104	127	116	105	
	024107	122	123	110	
	024112	111	120	040	
	024115	117	106	040	
	024120	124	104	122	
	024123	102	040	000	
2645	024126	015	012	104	ERR019: .ASCII <15><12>/DNI BIT FAILED TO SET AFTER /
	024131	116	111	040	
	024134	102	111	124	
	024137	040	106	101	
	024142	111	114	105	
	024145	104	040	124	
	024150	117	040	123	
	024153	105	124	040	
	024156	101	106	124	
	024161	105	122	040	
2646	024164	123	124	117	.ASCIZ /STOP PORT COMMAND/
	024167	120	040	120	
	024172	117	122	124	
	024175	040	103	117	
	024200	115	115	101	
	024203	116	104	000	
2647	024206	015	012	104	ERR020: .ASCII <15><12>/DATA COMPARE ERROR IN /
	024211	101	124	101	
	024214	040	103	117	
	024217	115	120	101	
	024222	122	105	040	
	024225	105	122	122	
	024230	117	122	040	
	024233	111	116	040	
2648	024236	124	122	101	.ASCIZ /TRANSMIT DESCRIPTOR RING/
	024241	116	123	115	
	024244	111	124	040	
	024247	104	105	123	
	024252	103	122	111	
	024255	120	124	117	
	024260	122	040	122	
	024263	111	116	107	
	024266	000			



2649	024267	015	012	104	ERR021: .ASCII <15><12>/DATA COMPARE ERROR IN /
	024272	101	124	101	
	024275	040	103	117	
	024300	115	120	101	
	024303	122	105	040	
	024306	105	122	122	
	024311	117	122	040	
	024314	111	116	040	
	024317	040			
2650	024320	122	105	103	.ASCIZ /RECEIVE DESCRIPTOR RING/
	024323	105	111	126	
	024326	105	040	104	
	024331	105	123	103	
	024334	122	111	120	
	024337	124	117	122	
	024342	040	122	111	
	024345	116	107	000	
2651	024350	015	012	124	ERR022: .ASCIZ <15><12>/TRANSMIT-RECEIVE DATA COMPARE ERROR /
	024353	122	101	116	
	024356	123	115	111	
	024361	124	055	122	
	024364	105	103	105	
	024367	111	126	105	
	024372	040	104	101	
	024375	124	101	040	
	024400	103	117	115	
	024403	120	101	122	
	024406	105	040	105	
	024411	122	122	117	
	024414	122	040	000	
2652	024417	015	012	103	ERR023: .ASCIZ <15><12>/CRC COMPARE ERROR /
	024422	122	103	040	
	024425	103	117	115	
	024430	120	101	122	
	024433	105	040	105	
	024436	122	122	117	
	024441	122	040	000	
2653	024444	015	012	111	ERR024: .ASCIZ <15><12>/INTERNAL ROM CRC COMPARE ERROR /
	024447	116	124	105	
	024452	122	116	101	
	024455	114	040	122	
	024460	117	115	040	
	024463	103	122	103	
	024466	040	103	117	
	024471	115	120	101	
	024474	122	105	040	
	024477	105	122	122	
	024502	117	122	040	
	024505	000			
2654	024506	015	012	122	ERR025: .ASCIZ <15><12>/RCBI BIT FAILED TO SET /
	024511	103	102	111	
	024514	040	102	111	
	024517	124	040	106	
	024522	101	111	114	
	024525	105	104	040	
	024530	124	117	040	
	024533	123	105	124	

	024536	040	000		
2655					
2656	024540				ERR026:
2657					
2658	024540	015	012	124	ERR027: .ASCII <15><12>/TIMEOUT ERROR - DELUA FAILED TO RELINQUISH/
	024543	111	115	105	
	024546	117	125	124	
	024551	040	105	122	
	024554	122	117	122	
	024557	040	055	040	
	024562	104	105	114	
	024565	125	101	040	
	024570	106	101	111	
	024573	114	105	104	
	024576	040	124	117	
	024601	040	122	105	
	024604	114	111	116	
	024607	121	125	111	
	024612	123	110		
2659	024614	040	117	127	.ASCIZ / OWNERSHIP OF FIRST TDRB/
	024617	116	105	122	
	024622	123	110	111	
	024625	120	040	117	
	024630	106	040	106	
	024633	111	122	123	
	024636	124	040	124	
	024641	104	122	102	
	024644	000			
2660	024645	015	012	124	ERR028: .ASCII <15><12>/TIMEOUT ERROR - DELUA FAILED TO RELINQUISH/
	024650	111	115	105	
	024653	117	125	124	
	024656	040	105	122	
	024661	122	117	122	
	024664	040	055	040	
	024667	104	105	114	
	024672	125	101	040	
	024675	106	101	111	
	024700	114	105	104	
	024703	040	124	117	
	024706	040	122	105	
	024711	114	111	116	
	024714	121	125	111	
	024717	123	110		
2661	024721	040	117	127	.ASCIZ / OWNERSHIP OF SECOND TDRB/
	024724	116	105	122	
	024727	123	110	111	
	024732	120	040	117	
	024735	106	040	123	
	024740	105	103	117	
	024743	116	104	040	
	024746	124	104	122	
	024751	102	000		
2662					
2663	024753				ERR029:
2664					
2665	024753	015	012	124	ERR030: .ASCII <15><12>/TIMEOUT ERROR - DELUA FAILED TO RELINQUISH/
	024756	111	115	105	

	024761	117	125	124	
	024764	040	105	122	
	024767	122	117	122	
	024772	040	055	040	
	024775	104	105	114	
	025000	125	101	040	
	025003	106	101	111	
	025006	114	105	104	
	025011	040	124	117	
	025014	040	122	105	
	025017	114	111	116	
	025022	121	125	111	
	025025	123	110		
2666	025027	040	117	127	.ASCIZ / OWNERSHIP OF FIRST RDRB/
	025032	116	105	122	
	025035	123	110	111	
	025040	120	040	117	
	025043	106	040	106	
	025046	111	122	123	
	025051	124	040	122	
	025054	104	122	102	
	025057	000			
2667	025060	015	012	124	ERR031: .ASCII <15><12>/TIMEOUT ERROR - DELUA FAILED TO RELINQUISH/
	025063	111	115	105	
	025066	117	125	124	
	025071	040	105	122	
	025074	122	117	122	
	025077	040	055	040	
	025102	104	105	114	
	025105	125	101	040	
	025110	106	101	111	
	025113	114	105	104	
	025116	040	124	117	
	025121	040	122	105	
	025124	114	111	116	
	025127	121	125	111	
	025132	123	110		
2668	025134	040	117	127	.ASCIZ / OWNERSHIP OF SECOND RDRB/
	025137	116	105	122	
	025142	123	110	111	
	025145	120	040	117	
	025150	106	040	123	
	025153	105	103	117	
	025156	116	104	040	
	025161	122	104	122	
	025164	102	000		
2669					
2670	025166				ERR032:
2671					
2672	025166	015	012	104	ERR033: .ASCII <15><12>/DATA COMPARE ERROR IN /
	025171	101	124	101	
	025174	040	103	117	
	025177	115	120	101	
	025202	122	105	040	
	025205	105	122	122	
	025210	117	122	040	
	025213	111	116	040	

2673	025216	106	111	122	.ASCIZ /FIRST TRANSMIT DESCRIPTOR RING/
	025221	123	124	040	
	025224	124	122	101	
	025227	116	123	115	
	025232	111	124	040	
	025235	104	105	123	
	025240	103	122	111	
	025243	120	124	117	
	025246	122	040	122	
	025251	111	116	107	
	025254	000			
2674	025255	015	012	104	ERR034: .ASCII <15><12>/DATA COMPARE ERROR IN /
	025260	101	124	101	
	025263	040	103	117	
	025266	115	120	101	
	025271	122	105	040	
	025274	105	122	122	
	025277	117	122	040	
	025302	111	116	040	
2675	025305	123	105	103	.ASCIZ /SECOND TRANSMIT DESCRIPTOR RING/
	025310	117	116	104	
	025313	040	124	122	
	025316	101	116	123	
	025321	115	111	124	
	025324	040	104	105	
	025327	123	103	122	
	025332	111	120	124	
	025335	117	122	040	
	025340	122	111	116	
	025343	107	000		
2676					
2677	025345				ERR035:
2678					
2679	025345	015	012	104	ERR036: .ASCII <15><12>/DATA COMPARE ERROR IN /
	025350	101	124	101	
	025353	040	103	117	
	025356	115	120	101	
	025361	122	105	040	
	025364	105	122	122	
	025367	117	122	040	
	025372	111	116	040	
2680	025375	106	111	122	.ASCIZ /FIRST RECEIVE DESCRIPTOR RING/
	025400	123	124	040	
	025403	122	105	103	
	025406	105	111	126	
	025411	105	040	104	
	025414	105	123	103	
	025417	122	111	120	
	025422	124	117	122	
	025425	040	122	111	
	025430	116	107	000	
2681	025433	015	012	104	ERR037: .ASCII <15><12>/DATA COMPARE ERROR IN /
	025436	101	124	101	
	025441	040	103	117	
	025444	115	120	101	
	025447	122	105	040	
	025452	105	122	122	

	025455	117	122	040	
	025460	111	116	040	
2682	025463	123	105	103	.ASCIZ /SECOND RECEIVE DESCRIPTOR RING/
	025466	117	116	104	
	025471	040	122	105	
	025474	103	105	111	
	025477	126	105	040	
	025502	104	105	123	
	025505	103	122	111	
	025510	120	124	117	
	025513	122	040	122	
	025516	111	116	107	
	025521	000			
2683	025522	015	012	104	ERR038: .ASCIZ <15><12>/DNI BIT NOT SET AFTER PORT HALT COMMAND /
	025525	116	111	040	
	025530	102	111	124	
	025533	040	116	117	
	025536	124	040	123	
	025541	105	124	040	
	025544	101	106	124	
	025547	105	122	040	
	025552	120	117	122	
	025555	124	040	110	
	025560	101	114	124	
	025563	040	103	117	
	025566	115	115	101	
	025571	116	104	040	
	025574	000			
2684	025575	015	012	105	ERR039: .ASCII <15><12>/ERROR - LOOPBACK SUCCESSFUL WITH/
	025600	122	122	117	
	025603	122	040	055	
	025606	040	114	117	
	025611	117	120	102	
	025614	101	103	113	
	025617	040	123	125	
	025622	103	103	105	
	025625	123	123	106	
	025630	125	114	040	
	025633	127	111	124	
	025636	110			
2685	025637	015	012	111	.ASCIZ <15><12>/INVALID DESTINATION ADDRESS /
	025642	116	126	101	
	025645	114	111	104	
	025650	040	104	105	
	025653	123	124	111	
	025656	116	101	124	
	025661	111	117	116	
	025664	040	101	104	
	025667	104	122	105	
	025672	123	123	040	
	025675	000			
2686	025676	015	012	106	ERR040: .ASCIZ <15><12>/FATAL ERROR - DELUA ID BIT NOT SET/
	025701	101	124	101	
	025704	114	040	105	
	025707	122	122	117	
	025712	122	040	055	
	025715	040	104	105	

	025720	114	125	101	
	025723	040	111	104	
	025726	040	102	111	
	025731	124	040	116	
	025734	117	124	040	
	025737	123	105	124	
	025742	000			
2687					
2688	025743	015	012	111	ERR041: .ASCIZ <15><12>/INTERNAL MEMORY DATA COMPARE ERROR /
	025746	116	124	105	
	025751	122	116	101	
	025754	114	040	115	
	025757	105	115	117	
	025762	122	131	040	
	025765	104	101	124	
	025770	101	040	103	
	025773	117	115	120	
	025776	101	122	105	
	026001	040	105	122	
	026004	122	117	122	
	026007	040	000		
2689					
2690	026011	015	012	104	ERR042: .ASCII <15><12>/DNI BIT FAILED TO SET AFTER /
	026014	116	111	040	
	026017	102	111	124	
	026022	040	106	101	
	026025	111	114	105	
	026030	104	040	124	
	026033	117	040	123	
	026036	105	124	040	
	026041	101	106	124	
	026044	105	122	040	
2691	026047	123	105	114	.ASCIZ /SELF TEST PORT COMMAND/
	026052	106	040	124	
	026055	105	123	124	
	026060	040	120	117	
	026063	122	124	040	
	026066	103	117	115	
	026071	115	101	116	
	026074	104	000		
2692					
2693	026076	015	012	047	ERR043: .ASCII <15><12>/'BUFL'.IN TDRB+6 NOT SET ON XMIT BUFF /
	026101	102	125	106	
	026104	114	047	054	
	026107	111	116	040	
	026112	124	104	122	
	026115	102	053	066	
	026120	040	116	117	
	026123	124	040	123	
	026126	105	124	040	
	026131	117	116	040	
	026134	130	115	111	
	026137	124	040	102	
	026142	125	106	106	
	026145	040			
2694	026146	117	126	105	.ASCIZ /OVERFLOW WITH <DTCR=0>/
	026151	122	106	114	

	026154	117	127	040	
	026157	127	111	124	
	026162	110	040	074	
	026165	104	124	103	
	026170	122	075	060	
	026173	076	000		
2695	026175	015	012	047	ERR044: .ASCII <15><12>/'BUFL' IN TDRB+6 NOT SET ON XM.T BUFF /
	026200	102	125	106	
	026203	114	047	040	
	026206	111	116	040	
	026211	124	104	122	
	026214	102	053	066	
	026217	040	116	117	
	026222	124	040	123	
	026225	105	124	040	
	026230	117	116	040	
	026233	130	115	111	
	026236	124	040	102	
	026241	125	106	106	
	026244	040			
2696	026245	117	126	105	.ASCIZ /OVERFLOW WITH <DTCR=1>/
	026250	122	106	114	
	026253	117	127	040	
	026256	127	111	124	
	026261	110	040	074	
	026264	104	124	103	
	026267	122	075	061	
	026272	076	000		
2697					
2698	026274	015	012	120	ERR045: .ASCIZ <15><12>/PCSR0 INTERRUPT BIT CLEAR ERROR /
	026277	103	123	122	
	026302	060	040	111	
	026305	116	124	105	
	026310	122	122	125	
	026313	120	124	040	
	026316	102	111	124	
	026321	040	103	114	
	026324	105	101	122	
	026327	040	105	122	
	026332	122	117	122	
	026335	040	000		
2699					
2700	026337	015	012	122	ERR046: .ASCIZ <15><12>/RECEIVED PACKET COUNTER NOT GREATER THAN 0 /
	026342	105	103	105	
	026345	111	126	105	
	026350	104	040	120	
	026353	101	103	113	
	026356	105	124	040	
	026361	103	117	125	
	026364	116	124	105	
	026367	122	040	116	
	026372	117	124	040	
	026375	107	122	105	
	026400	101	124	105	
	026403	122	040	124	
	026406	110	101	116	
	026411	040	060	040	

2701	026414	000							
	026415	015	012	123	ERR047: .ASCIZ <15><12>/SPARE /				
	026420	120	101	122					
	026423	105	040	000					
2702	026426	015	012	106	ERR048: .ASCIZ <15><12>/FATAL BIT SET DUE TO DEVICE OR UNIBUS ERROR /				
	026431	101	124	101					
	026434	114	040	102					
	026437	111	124	040					
	026442	123	105	124					
	026445	040	104	125					
	026450	105	040	124					
	026453	117	040	104					
	026456	105	126	111					
	026461	103	105	040					
	026464	117	122	040					
	026467	125	116	111					
	026472	102	125	123					
	026475	040	105	122					
	026500	122	117	122					
2703	026503	040	000		.EVEN				



```
2705 .SBTTL GLOBAL MACRO AND SUBROUTINES SECTION
2706
2707 ;*****
2708 ;
2709 ; MACRO FTL
2710 ;
2711 ; THIS MACRO CALLS SUBROUTINE 'CHKFTL'
2712 ;
2713 ; CALL: FTL
2714 ;
2715 ;*****
2716
2717 .MACRO FTL
2718
2719 .NLIST
2720 .LIST ME
2721 .LIST
2722
2723 JSR PC.CHKFTL ; 'FATL' BIT SET?
2724
2725 .NLIST ME
2726 .ENDM
2727
```

2729  
2730  
2731  
2732  
2733  
2734  
2735  
2736  
2737  
2738  
2739  
2740  
2741  
2742  
2743  
2744  
2745  
2746  
2747  
2748  
2749  
2750  
2751  
2752  
2753  
2754  
2755

```
*****  
:  
: MACRO PNTMAC  
:  
: THIS MACRO WILL SETUP AND CALL SUBROUTINE 'PNTID',  
: WHICH WILL THEN DISPLAY TEST NUMBER AND NAME.  
:  
: CALL: PNTMAC tname  
:  
: WHERE 'tname' IS THE POINTER TO THE  
: TEST NAME MESSAGE.  
:  
*****  
.  
.MACRO PNTMAC TNAME  
.  
.NLIST  
.LIST ME  
.LIST  
:  
MOV @TNAME,R4 ;GET POINTER TO TEST NAME MESSAGE  
JSR PC,PNTID ;PRINT TEST NUMBER AND NAME  
:  
END OF MACRO EXPANSION OF 'PNTMAC'  
.  
.NLIST ME  
.ENDM
```

```

2757
2758
2759
2760
2761
2762
2763
2764
2765
2766
2767
2768
2769
2770
2771
2772
2773
2774
2775
2776
2777
2778 026506
2779 026506 010146
2780 026510 010246
2781 026512 010546
2782 026514
2783 026514 112105
2784 026516 004737 031160'
2785 026522
2786 026522 077204
2787
2788 026524 005103
2789 026526 005104
2790
2791 026530 012702 016574'
2792 026534 010422
2793 026536 010322
2794
2795 026540 012605
2796 026542 012602
2797 026544 012601
2798 026546 000207
2799
    ;*****
    ;
    ; SUBROUTINE - BLKCRC
    ;
    ; THIS ROUTINE PERFORMS A CRC CALCULATION ON A BLOCK OF DATA
    ;
    ; THIS ROUTINE USED FOR ALL CRC CALCULATIONS EXCEPT ROM.
    ;
    ; INPUTS: R1 CONTAINS BASE ADDRESS OF DATA BLOCK
    ;          R2 CONTAINS DATA BLOCK BYTE COUNT
    ;          R3,R4 CONTAINS INITIAL CRC
    ;
    ; OUTPUT: R3,R4 CONTAIN CRC CODE
    ;
    ; CALLING SEQUENCE:      MOV     #_,R1      ;GET BASE ADDRESS
    ;                       MOV     #_,R2      ;GET BYTE COUNT
    ;                       JSR     PC,BLKCRC   ;CALCULATE CRC
    ;*****
    BLKCRC:
    MOV     R1,-(SP)      ;SAVE R1
    MOV     R2,-(SP)      ;SAVE R2
    MOV     R5,-(SP)      ;SAVE R5
    1$:
    MOVB   (R1)+,R5      ;GET NEXT BYTE
    JSR    PC,GETCRC     ;CALCULATE THE CRC
    2$:
    SOB    R2,1$        ;LOOP TILL DONE
    COM    R3            ; COMPLIMENT
    COM    R4            ; RESULTS
    MOV    #XCRC,R2     ; BASE ADDRESS OF SAVED CRC
    MOV    R4,(R2)+     ; SAVE 1ST WORD
    MOV    R3,(R2)+     ; SAVE 2ND WORD
    MOV    (SP)+,R5     ;RESTORE R5
    MOV    (SP)+,R2     ;RESTORE R2
    MOV    (SP)+,R1     ;RESTORE R1
    RTS    PC           ;RETURN TO CALLING ROUTINE
    
```

2801  
 2802  
 2803  
 2804  
 2805  
 2806  
 2807  
 2808  
 2809  
 2810  
 2811  
 2812  
 2813  
 2814  
 2815  
 2816  
 2817  
 2818  
 2819  
 2820  
 2821  
 2822  
 2823  
 2824  
 2825  
 2826  
 2827  
 2828  
 2829  
 2830  
 2831  
 2832  
 2833  
 2834  
 2835  
 2836  
 2837  
 2838  
 2839  
 2840  
 2841  
 2842  
 2843  
 2844  
 2845  
 2846

```

*****
:
: SUBROUTINE - CHKDNI
:
: THIS ROUTINE WAITS FOR DNI TO SET.
:
: INPUTS:          NONE
:
: OUTPUTS:         IF DNI SETS
:                   THEN CARRY = 0
:
:                   IF DNI FAILS TO SET
:                   THEN CARRY = 1
:                   PCSRO -> EPSCRO
:                   PCSR1 -> EPCSR1
:
: CALLING SEQUENCE:
: JSR      PC,CHKDNI
*****
    
```

```

CHKDNI:
MOV     R0,-(SP)      ; SAVE R0
MOV     R1,-(SP)      ; SAVE R1
MOV     R4,-(SP)      ; SAVE R4
MOV     #600,METER    ; PUT SOME TIME IN THE TIMER
JSR     PC,TIMON      ; TURN ON THE LINE CLOCK
10$:   MOV     @PCSRO,R4 ; GET PCSRO
        BIT     @DNI,R4  ; IS DNI SET?
        BNE     30$     ; YES
        BREAK      ; NO, VISIT DRS FOR A MOMENT
                                TRAP      C$BRK
        TST     METER    ; HAS TIMER EXPIRED?
        BNE     10$     ; NOT YET
        MOV     R4,EPCSR0 ; PCSRO -> EPCSR0
        MOV     @PCSR1,EPCSR1 ; PCSR1 -> EPCSR1
        JSR     PC,TIMOFF ; TURN OFF THE TIMER
        SEC     ; SET CARRY
        BR      40$
30$:   JSR     PC,TIMOFF ; TURN OFF THE TIMER
        CLC     ; DNI SET SO CLEAR CARRY
40$:   MOV     (SP)+,R4  ; RESTORE R4
        MOV     (SP)+,R1 ; RESTORE R1
        MOV     (SP)+,R0 ; RESTORE R0
        RTS     PC      ; AND RETURN
    
```

2848  
 2849  
 2850  
 2851  
 2852  
 2853  
 2854  
 2855  
 2856  
 2857  
 2858  
 2859  
 2860  
 2861

```

*****
SUBROUTINE - CHKFTL
THIS SUBROUTINE WILL CHECK FOR FATAL ERROR BIT SET
IF SET, WILL ISSUE MESSAGE TO IGNORE CONTENTS OF PCSR1
INPUTS: NONE
OUTPUTS: IF 'FATL' SET, MESSAGE PRINTED
CALL: JSR PC,CHKFTL
*****
    
```

2862 026652  
 2863 026652  
 2864 026654  
 2865 026660  
 2866 026664  
 2867 026666  
 2868 026706  
 2869 026714  
 2870 026722  
 2871 026724  
 2872

010046  
 017700 151344  
 032700 001000  
 001410  
 012746 026726'  
 012746 000001  
 010600  
 104417  
 062706 000004  
 017737 151312 016514'  
 017737 151306 016516'  
 012600  
 000207

```

CHKFTL:
MOV    R0,-(SP)      ;SAVE R0
MOV    @PCSR0,R0     ;GET CONTENTS OF CSRO
BIT    @FATL,R0      ;CUNTENTS OF PCSR1 VALID?
BEQ    1$            ;YES, EXIT
PRINTF @FTLSET
MOV    @FTLSET,-(SP)
MOV    @1,-(SP)
MOV    SP,R0
TRAP   C$PNTF
ADD    @4,SP

1$:   MOV    @PCSR0,EPCSR0 ;SAVE CONTENTS OF PCSRO
      MOV    @PCSR1,EPCSR1 ;SAVE CONTENTS OF PCSR1
      MOV    (SP)+,R0      ;RESTORE R0
      RTS    PC            ;RETURN TO CALLING ROUTINE
    
```

2873  
 2874

026726 045 116 045  
 026731 101 047 106  
 026734 101 124 114  
 026737 047 040 102  
 026742 111 124 040  
 026745 123 105 124  
 026750 040 055 040  
 026753 104 101 124  
 026756 101 040 111  
 026761 116 040 120  
 026764 103 123 122  
 026767 061 040 116  
 026772 117 124 040  
 026775 126 101 114  
 027000 111 104 040  
 027003 106 117 122  
 027006 040 124 110  
 027011 111 123 040  
 027014 105 122 122  
 027017 117 122 056  
 027022 000

FTLSET:.asciz/%N%A'FATL' BIT SET - DATA IN PCSR1 NOT VALID FOR THIS ERROR./

.even

```

2876 :*****
2877 :
2878 :       SUBROUTINE - CHKOWN
2879 :
2880 :       THIS ROUTINE CHECKS FOR THE OWNERSHIP BIT IN
2881 :       BOTH TRANSMIT AND RECEIVE DESCRIPTOR RINGS.
2882 :
2883 :       INPUTS:           RS = ADDRESS OF DESCRIPTOR RING
2884 :
2885 :       OUTPUTS:          IF OWN BIT = 0 (PORT DRIVER)
2886 :                           THEN CARRY = 0
2887 :
2888 :                           IF OWN BIT = 1 (UNA)
2889 :                           THEN CARRY = 1
2890 :
2891 :*****
    
```

```

2893 027024  CHKOWN:
2894 027024  010046  MOV     R0,-(SP)           ; SAVE R0
2895 027026  010446  MOV     R4,-(SP)           ; SAVE R4
2896 027030  012704  000010  MOV     #10,R4            ; DELAY VALUE
2897 027034  004737  033416' JSR     PC,TIMON           ; TURN ON CLOCK
2898 027040  016500  000004  1$:    MOV     4(R5),R0        ; GET TRDB+4
2899 027044  032700  100000  BIT     #OWN,R0           ; BIT15, OWNERSHIP SET?
2900 027050  001406  BEQ     10$                ; NO, EXIT ROUTINE
2901 027052  104422  BREAK                    ; VISIT DRS WHILE WAITING
2902 027054  005737  016602' TST     METER              ; TIME UP?
2903 027060  001367  BNE     1$                 ; NO, LOOP AGAIN
2904 027062  000261  SEC                    ; YES, SET CARRY = 1
2905 027064  000401  BR      20$                ; GET OUT
2906
2907 027066  000241  10$:   CLC                    ; CLEAR CARRY
2908 027070  20$:
2909 027070  004737  033402' JSR     PC,TIMOFF          ; TURN OFF TIMER
2910 027074  012604  MOV     (SP)+,R4          ; RESTORE R4
2911 027076  012600  MOV     (SP)+,R0          ; RESTORE R0
2912 027100  000207  RTS     PC                 ; AND RETURN
    
```

```

2914
2915 *****
2916 :
2917 : SUBROUTINE  CHKRCCE
2918 :
2919 : THIS ROUTINE WAITS FOR RCBI TO SET.
2920 :
2921 : INPUTS:      NONE
2922 :
2923 : OUTPUTS:     IF RCBI SFTS
2924 :               THEN CARRY = 0
2925 :
2926 :               IF RCBI FAILS TO SET
2927 :                 THEN CARRY = 1
2928 :                 PCSRO -> EPSCRO
2929 :                 PCSR1 -> EPCSR1
2930 :
2931 : CALLING SEQUENCE:
2932 : JSR      PC,CHKRCE
2933 :
2934 *****
    
```

```

2935
2936 027102          CHKRCE:
2937 027102 010046      MOV      R0,-(SP)          ; SAVE R0
2938 027104 010146      MOV      R1,-(SP)          ; SAVE R1
2939 027106 012737 000473 016602'  MOV      @5*SECOND,METER      ; PUT SOME TIME IN THE TIMER
2940 027114 004737 033416'      JSR      PC,TIMON           ; TURN ON THE LINE CLOCK
2941 027120 017737 151100 000236' 10$: MOV      @PCSRO,PCSROC        ; GET PCSRO
2942 027126 032737 002000 000236' BIT      @RCBI,PCSROC        ; IS RCBI SET?
2943 027134 001016      BNE      30$              ; YES
2944 027136          BRF      BRF      30$              ; NO, VISIT DRS FOR A MOMENT
2945 027140 005737 016602'      TS      METER              ; HAS TIMER EXPIRED?
2946 027144 001365      BNE      10$              ; NOT YET
2947 027146 013737 000236' 016514' MOV      PCSROC,EPCSR0        ; PCSRO -> EPCSR0
2948 027154 017737 151046 016516' MOV      @PCSR1,EPCSR1        ; PCSR1 -> EPCSR1
2949 027162 004737 033402'      JSR      PC,TIMOFF          ; TURN OFF THE TIMER
2950 027166 000261      SEC                      ; SET CARRY
2951 027170 000403      BR      40$
2952 027172 004737 033402' 30$: JSR      PC,TIMOFF          ; TURN OFF THE TIMER
2953 027176 000241      CLC                      ; RCBI SET SO CLEAR CARRY
2954 027200          40$:
2955 027200 012601      MOV      (SP)+,R1          ; RESTORE R1
2956 027202 012600      MOV      (SP)+,R0          ; RESTORE R0
2957 027204 000207      RTS      PC              ; AND RETURN
    
```

2959  
 2960  
 2961  
 2962  
 2963  
 2964  
 2965  
 2966  
 2967  
 2968  
 2969  
 2970  
 2971  
 2972  
 2973  
 2974  
 2975  
 2976  
 2977  
 2978  
 2979  
 2980  
 2981  
 2982  
 2983  
 2984  
 2985  
 2986  
 2987  
 2988  
 2989

```

*****
SUBROUTINE - CHKRDR
THIS SUBROUTINE COMPARES A RECEIVE DESCRIPTOR RING ENTRY
WITH EXPECTED DATA.
INPUTS:          R5 = ADDRESS OF RDRB TO BE COMPARED.
INPLICIT INPUTS: XRDRB0 = TABLE WITH EXPECTED DATA
OUTPUTS:         IF COMPARE IS SUCCESSFUL
                  THEN CARRY = 0
                  IF COMPARE IS UNSUCCESSFUL
                  THEN CARRY = 1
                  EXPECTED RDRB+0 = XRDRB0
                  EXPECTED RDRB+2 = XRDRB2
                  EXPECTED RDRB+4 = XRDRB4
                  EXPECTED RDRB+6 = XRDRB6
                  ACTUAL RDRB+0  -> ERDRB0
                  ACTUAL RDRB+2  -> ERDRB2
                  ACTUAL RDRB+4  -> ERDRB4
                  ACTUAL RDRB+6  -> ERDRB6
CALLING SEQUENCE:
                  JSR      PC,CHKRDR
*****
    
```

2990 027206  
 2991 027206 010046  
 2992 027210 010146  
 2993 027212 010346  
 2994 027214 010446  
 2995 027216 012700 000004  
 2996 027222 012703 016530'  
 2997 027226 010504  
 2998 027230  
 2999 027230 022324  
 3000 027232 001012  
 3001 027234 005300  
 3002 027236 001374  
 3003  
 3004 027240 011400  
 3005 027242 042700 007777  
 3006 027246 011301  
 3007 027250 042701 007777  
 3008 027254 020001  
 3009 027256 001411  
 3010 027260  
 3011 027260 012703 016520'  
 3012 027264 010504  
 3013 027266 012423  
 3014 027270 012423  
 3015 027272 012423

```

CHKRDR:
MOV      R0, -(SP)      ; SAVE R0
MOV      R1, (SP)      ; SAVE R1
MOV      R3, -(SP)      ; SAVE R3
MOV      R4, -(SP)      ; SAVE R4
MOV      #4, R0         ; DO FOUR COMPARES
MOV      #XRDRB0, R3    ; R3 POINTS TO EXPECTED DATA
MOV      R5, R4         ; R4 POINTS TO ACTUAL RDRB
10$:
CMP      (R3)+, (R4)+   ; ERROR IN ACTUAL TABLE DATA?
BNE      20$           ; YES
DEC      R0             ; REDUCE LOOP COUNT
BNE      10$           ; IF NOT FINISHED, LOOP AGAIN
MOV      (R4), R0       ; RDRB+6 -> R0
BIC      #TDRMSK, R0    ; MASK OUT TDR VALUE
MOV      (R3), R1       ; GET EXPECTED
BIC      #TDRMSK, R1    ; MASK OUT TDR VALUE
CMP      R0, R1         ; COMPARE ERROR ?
BEQ      30$           ; YES
20$:
MOV      #ERDRB0, R3    ; R3 POINTS TO ACTUAL TABLE
MOV      R5, R4         ; R4 POINTS TO ACTUAL RDRB
MOV      (R4)+, (R3)+   ; LOAD ACTUAL TABLE
MOV      (R4)+, (R3)+
MOV      (R4)+, (R3)+
    
```



3016	027274	012423	MOV	(R4)+,(R3)+	
3017	027276	000261	SEC		; SET CARRY
3018	027300	000401	BR	40\$	
3019	027302	000241	CLC		; CLEAR CARRY
3020	027304	012604	MOV	(SP)+,R4	; RESTORE R4
3021	027306	012603	MOV	(SP)+,R3	; RESTORE R3
3022	027310	012601	MOV	(SP)+,R1	; RESTORE R1
3023	027312	012600	MOV	(SP)+,R0	; RESTORE R0
3024	027314	000207	RTS	PC	; AND RETURN

```

3026
3027
3028
3029
3030
3031
3032
3033
3034
3035
3036
3037
3038
3039
3040
3041
3042
3043
3044
3045
3046
3047
3048 027316
3049 027316 010046
3050 027320 010146
3051 027322 010446
3052 027324 012737 000176 016602'
3053 027332 004737 033416'
3054 027336 017704 150662 10$:
3055 027342 032704 020000
3056 027346 001015
3057 027350
    027350 104422
3058 027352 005737 016602'
3059 027356 001367
3060 027360 010437 016514'
3061 027364 017737 150636 016516'
3062 027372 004737 033402'
3063 027376 000261
3064 027400 000403
3065 027402 004737 033402' 30$:
3066 027406 000241
3067 027410 012604 40$:
3068 027412 012601
3069 027414 012600
3070 027416 000207
    *****
    ;
    ; SUBROUTINE - CHKRXI
    ;
    ; THIS ROUTINE WAITS FOR RXI TO SET.
    ;
    ; INPUTS: NONE
    ;
    ; OUTPUTS: IF RXI SETS
    ;           THEN CARRY = 0
    ;
    ;           IF RXI FAILS TO SET
    ;           THEN CARRY = 1
    ;           PCSRO -> EPCSRO
    ;           PCSR1 -> EPCSR1
    ;
    ; CALLING SEQUENCE:
    ; JSR PC,CHKRXI
    ;
    *****
CHKRXI:
    MOV R0,-(SP) ; SAVE R0
    MOV R1,-(SP) ; SAVE R1
    MOV R4,-(SP) ; SAVE R4
    MOV #2*SECOND,METER ; PUT SOME TIME IN THE TIMER
    JSR PC,TIMON ; TURN ON THE LINE CLOCK
    MOV @PCSRO,R4 ; GET PCSRO
    BIT @RXI,R4 ; IS RXI SET?
    BNE 30$ ; YES
    BREAK ; NO, VISIT DRS FOR A MOMENT TRAP C$BRK
    TST METER ; HAS TIMER EXPIRED?
    BNE 10$ ; NOT YET
    MOV R4,EPCSRO ; PCSRO -> EPCSRO
    MOV @PCSR1,EPCSR1 ; PCSR1 -> EPCSR1
    JSR PC,TIMOFF ; TURN OFF THE TIMER
    SEC ; SET CARRY
    BR 40$
    JSR PC,TIMOFF ; TURN OFF THE TIMER
    CLC ; RXI SET SO CLEAR CARRY
    MOV (SP)+,R4 ; RESTORE R4
    MOV (SP)+,R1 ; RESTORE R1
    MOV (SP)+,R0 ; RESTORE R0
    RTS PC ; AND RETURN
    
```

3072  
 3073  
 3074  
 3075  
 3076  
 3077  
 3078  
 3079  
 3080  
 3081  
 3082  
 3083  
 3084  
 3085  
 3086  
 3087  
 3088  
 3089  
 3090  
 3091  
 3092  
 3093 027420  
 3094 027420 010046  
 3095 027422 010446  
 3096 027424 017704 150576  
 3097 027430 042704 140377  
 3098 027434 022704 000000  
 3099 027440 001413  
 3100  
 3101  
 3102 027442 042704 140377  
 3103 027446 012700 000010  
 3104 027452 006204  
 3105 027454 005300  
 3106 027456 001375  
 3107 027460 010437 016600'  
 3108 027464 000261  
 3109 027466 000401  
 3110 027470 000241  
 3111 027472 012604  
 3112 027474 012600  
 3113 027476 000207

```

*****
:
: SUBROUTINE CHKSTR
:
: THIS TEST CHECKS THE SELF TEST RESULTS.
:
: INPUTS: NONE
:
: OUTPUTS: IF SELF TEST SUCCESSFUL
:          THEN CARRY = 0
:
:          IF SELF TEST FAILED
:          THEN CARRY = 1
:          SELF TEST CODE SHIFTED RIGHT -> ECODE
:
: CALLING SEQUENCE:
:          JSR PC,CHKSTR
*****
    
```

```

CHKSTR:
    MOV R0,-(SP) ; SAVE R0
    MOV R4,-(SP) ; SAVE R4
    MOV @PCSR1,R4 ; PCSR1 -> R4
    BIC @STMASK,R4 ; MASK SELF TEST CODE BITS
    CMP @GOODST,R4 ; SELF TEST SUCCESSFUL ?
    BEQ 10$ ; YES
;
; SELF TEST FAILED
    BIC @STMASK,R4
    MOV @8.,R0 ; SHIFT CODE RIGHT
5$: ASR R4
    DEC R0
    BNE 5$
    MOV R4,ECODE ; SHIFTED CODE -> ECODE
    SEC ; SET CARRY
    BR 20$
10$: CLC ; SELF TEST PASSED CLEAR CARRY
20$: MOV (SP)+,R4 ; RESTORE R4
    MOV (SP)+,R0 ; RESTORE R0
    RTS PC ; AND RETURN
    
```

```

3115 *****
3116 :
3117 : SUBROUTINE - CHKTDR
3118 :
3119 : THIS SUBROUTINE COMPARES A TRANSMIT DESCRIPTOR RING ENTRY
3120 : WITH EXPECTED DATA.
3121 :
3122 : INPUTS: R5 = ADDRESS OF TDRB TO BE COMPARED
3123 :
3124 : IMPLICIT INPUTS:
3125 : XTDRB0 = TABLE WITH EXPECTED DATA
3126 :
3127 : OUTPUTS: IF COMPARE IS SUCCESSFUL
3128 : THEN CARRY = 0
3129 :
3130 : IF COMPARE IS UNSUCCESSFUL
3131 : THEN CARRY = 1
3132 : EXPECTED TDRB+0 = XTDRB0
3133 : EXPECTED TDRB+2 = XTDRB2
3134 : EXPECTED TDRB+4 = XTDRB4
3135 : EXPECTED TDRB+6 = XTDRB6
3136 : ACTUAL TDRB+0 -> ETDRB0
3137 : ACTUAL TDRB+2 -> ETDRB2
3138 : ACTUAL TDRB+4 -> ETDRB4
3139 : ACTUAL TDRB+6 -> ETDRB6
3140 :
3141 : CALLING SEQUENCE:
3142 : JSR PC,CHKTDR
3143 :
3144 *****
    
```

```

3145 027500 CHKTDR:
3146 027500 010046 MOV R0,-(SP) ; SAVE R0
3147 027502 010346 MOV R3,-(SP) ; SAVE R3
3148 027504 010446 MOV R4,-(SP) ; SAVE R4
3149 027506 012700 000004 MOV #4,R0 ; DO FOUR COMPARES
3150 027512 012703 016550' MOV #XTDRB0,R3 ; R3 POINTS TO EXPECTED DATA
3151 027516 010504 MOV R5,R4 ; R4 POINTS TO ACTUAL TDRB
3152 027520 10$: CMP (R3)+,(R4)+ ; ERROR IN ACTUAL TABLE DATA?
3153 027520 022324 BNE 20$ ; YES
3154 027522 001003 DEC ,0 ; REDUCE LOOP COUNT
3155 027524 005300 BNE 10$ ; IF NOT FINISHED, LOOP AGAIN
3156 027526 001374 BR 30$
3157 027530 000411
3158 027532 012703 016540' 20$: MOV #ETDRB0,R3 ; R3 POINTS TO ACTUAL TABLE
3159 027536 010504 MOV R5,R4 ; R4 POINTS TO ACTUAL TDRB
3160 027540 012423 MOV (R4)+,(R3)+ ; LOAD ACTUAL TABLE
3161 027542 012423 MOV (R4)+,(R3)+
3162 027544 012423 MOV (R4)+,(R3)+
3163 027546 012423 MOV (R4)+,(R3)+
3164 027550 000261 SEC ; SET CARRY
3165 027552 000401 BR 40$
3166 027554 000241 30$: CLC ; CLEAR CARRY
3167 027556 012604 40$: MOV (SP)+,R4 ; RESTORE R4
3168 027560 012603 MOV (SP)+,R3 ; RESTORE R3
3169 027562 012600 MOV (SP)+,R0 ; RESTORE R0
3170 027564 000207 RTS PC ; AND RETURN
    
```

3172  
 3173  
 3174  
 3175  
 3176  
 3177  
 3178  
 3179  
 3180  
 3181  
 3182  
 3183  
 3184  
 3185  
 3186  
 3187  
 3188  
 3189  
 3190  
 3191  
 3192  
 3193  
 3194  
 3195  
 3196  
 3197  
 3198  
 3199  
 3200  
 3201  
 3202  
 3203  
 3204  
 3205  
 3206  
 3207  
 3208  
 3209  
 3210  
 3211  
 3212  
 3213  
 3214  
 3215  
 3216

```

*****
:
: SUBROUTINE - CHKTXI
:
: THIS ROUTINE WAITS FOR TXI TO SET.
:
: INPUTS:      NONE
:
: OUTPUTS:     IF TXI SETS
:                THEN CARRY = 0
:
:                IF TXI FAILS TO SET
:                THEN CARRY = 1
:                PCSRO -> EPSCRO
:                PCSR1 -> EPCSR1
:
: CALLING SEQUENCE:
:                JSR      PC,CHKTXI
*****
    
```

```

CHKTXI:
MOV      R0,-(SP)      ; SAVE R0
MOV      R1,-(SP)      ; SAVE R1
MOV      R4,-(SP)      ; SAVE R4
MOV      #2*SECOND,METER ; PUT SOME TIME IN THE TIMER
JSR      PC,TIMON      ; TURN ON THE LINE CLOCK
10$:    MOV      @PCSRO,PCSROC ; GET PCSRO
BIT      #TXI,PCSROC   ; IS TXI SET?
BNE      30$           ; YES
BREAK    ; NO, VISIT DRS FOR A MOMENT
                                TRAP    C$BRK
3204:   TST      METER      ; HAS TIMER EXPIRED?
BNE      10$           ; NOT YET
MOV      PCSROC,EPCSR0 ; PCSRO -> EPCSR0
MOV      @PCSR1,EPCSR1 ; PCSR1 -> EPCSR1
JSR      PC,TIMOFF     ; TURN OFF THE TIMER
SEC      ; SET CARRY
BR       40$
30$:    JSR      PC,TIMOFF ; TURN OFF THE TIMER
CLC      ; TXI SET SO CLEAR CARRY
40$:    MOV      (SP)+,R4  ; RESTORE R4
MOV      (SP)-,R1      ; RESTORE R1
MOV      (SP)+,R0      ; RESTORE R0
RTS      PC            ; AND RETURN
    
```

027566  
 027570 010046  
 027572 010146  
 027572 010446  
 027574 000176 016602'  
 027602 004737 033416'  
 027606 017737 150412 000236' 10\$:  
 027614 032737 010000 000236'  
 027622 001016  
 027624 104422  
 027626 005737 016602'  
 027632 001365  
 027634 013737 000236' 016514'  
 027642 017737 150360 016516'  
 027650 004737 033402'  
 027654 000261  
 027656 000403  
 027660 004737 033402' 30\$:  
 027664 000241  
 027666 012604 40\$:  
 027670 012601  
 027672 012600  
 027674 000207

3218  
 3219  
 3220  
 3221  
 3222  
 3223  
 3224  
 3225  
 3226  
 3227  
 3228  
 3229  
 3230  
 3231  
 3232  
 3233  
 3234  
 3235  
 3236  
 3237  
 3238  
 3239  
 3240  
 3241  
 3242  
 3243  
 3244  
 3245  
 3246  
 3247  
 3248

027676 010446  
 027700 017704 150320  
 027704 032704 100000  
 027710 001007  
 027712 010437 016514'  
 027716 017737 150304 016516'  
 027724 000261  
 027726 000401  
 027730 000241  
 027732 012604  
 027734 000207

```

*****
:
: SUBROUTINE CHKSER
:
: THIS ROUTINE CHECKS FOR THE SERI BIT IN PCSRO.
:
: INPUTS: NONE
:
: OUTPUTS: IF SERI BIT SET THEN CARRY = 0
:
: IF SERI BIT NOT SET THEN CARRY = 1
: PCSRO -> EPCSRO
: PCSR1 -> EPCSR1
:
: CALLING SEQUENCE:
: JSR PC,CHKSER
*****
CHKSER:MOV R4,-(SP) ;SAVE R4
MOV @PCSRO,R4 ;GET PCSRO CONTENTS
BIT @SERI,R4 ;IS SERI BIT SET?
BNE 10$ ;YES
MOV R4,EPCSRO ;NO. SAVE PCSRO CONTENTS
MOV @PCSR1,EPCSR1 ;GET PCSR1 CONTENTS TOO
SEC ;INDICATE NO SERI BIT
BR 20$ ;LEAVE
10$: CLC ;INDICATE SERI BIT SET
20$: MOV (SP)+,R4 ;RESTORE R4
RTS PC
    
```

```

3250
3251
3252
3253
3254
3255
3256
3257
3258
3259
3260
3261
3262
3263
3264
3265
3266
3267
3268
3269
3270
3271
3272
3273
3274
3275
3276 027736
3277 027736 010046
3278 027740 010146
3279 027742 010446
3280 027744 012737 001661 016602'
3281 027752 004737 033416'
3282 027756
3283 027756 017704 150242
3284 027762 032704 004000
3285 027766 001025
3286 027770
    027770 104422
3287 027772 005737 016602'
3288 027776 001367
3289
3290
3291
3292 030000 004737 033402'
3293 030004
    030004 012746 030060'
    030010 012746 000001
    030014 010600
    030016 104417
    030020 062706 000004
3294
3295 030024 010437 016514'
3296 030030 017737 150172 016516'
3297 030036 000261
3298 030040 000403
3299
3300 030042
    
```

```

:*****
:
:      SUBROUTINE - CKDNI
:
:      THIS SUBROUTINE WAITS FOR DONE INTERRUPT (DNI)TO SET.
:      IF A DNI IS RECEIVED BEFORE TIMER EXPIRES, PROCEED OK. IF
:      TIMER EXPIRES PRIOR TO AN INTERRUPT, OR THE INTERRUPT WAS
:      NOT CAUSED BY A DNI, THEN THE APPROPRIATE ERROR MESSAGE IS
:      ISSUED.
:
:      INPUTS:          NONE
:
:      OUTPUTS:        IF DNI SETS PRIOR TO TIMER TIME OUT
:                      THEN CARRY BIT = 0
:
:                      ELSE
:                      CARRY BIT = 1
:                      PCSRO -> EPSCRO
:                      PCSR1 -> EPSCR1
:
:      CALLING SEQUENCE:
:                      JSR      PC,CKDNI
:*****
    
```

```

CKDNI:
    MOV      R0,-(SP)          ; SAVE R0
    MOV      R1,-(SP)          ; SAVE R1
    MOV      R4,-(SP)          ; SAVE R4
    MOV      #15,*SECOND,METER ; PUT ENOUGH TIME ON TIMER
    JSR      PC,TIMON          ; TURN ON THE LINE CLOCK
10$:
    MOV      @PCSR0,R4         ; READ AND SAVE CONTENTS OF PCSRO
    BIT      #DNI,R4           ; DID WE GET A DNI INTERRUPT?
    BNE      20$               ; YES EXIT DELAY LOOP
    BREAK                                ; NO, VISIT DRS FOR A MOMENT
                                TRAP      C$BRK
    TST      METER              ; HAS TIMER EXPIRED?
    BNE      10$               ; NOT YET
;TIMER EXPIRED BEFORE DNI SET
    JSR      PC,TIMOFF          ; TURN OFF THE TIMER
    PRINTF   #INTMG1           ; PRINT TIMED OUT MESSAGE
                                MOV      #INTMG1,(SP)
                                MOV      #1,-(SP)
                                MOV      SP,R0
                                TRAP    C$PNTF
                                ADD     #4,SP
    MOV      R4,EPCSR0         ; PCSRO > EPCSR0
    MOV      @PCSR1,EPCSR1     ; PCSR1 -> EPCSR1
    SEC                                ; SET CARRY BIT
    BR      40$                ; GO EXIT
20$:
    
```

```
3301 030042 004737 033402'          JSR      PC,TIMOFF          ; TURN OFF THE TIMER
3302 030046 000241                   CLC                          ; DNI SET, SO CLEAR C BIT
3303 030050                   40$:
3304 030050 012604                   MOV      (SP)+,R4           ; RESTORE R4
3305 030052 012601                   MOV      (SP)+,R1           ; RESTORE R1
3306 030054 012600                   MOV      (SP)+,R0           ; RESTORE R0
3307 030056 000207                   RTS      PC                  ; AND RETURN
3308
3309 030060      045      116      045 INTMG1:.ASCIZ/!N!A DNI DID NOT SET PRIOR TO SOFTWARE TIMER TIME OUT./
      030063      101      040      104
      030066      116      111      040
      030071      104      111      104
      030074      040      116      117
      030077      124      040      123
      030102      105      124      040
      030105      120      122      111
      030110      117      122      040
      030113      124      117      040
      030116      123      117      106
      030121      124      127      101
      030124      122      105      040
      030127      124      111      115
      030132      105      122      040
      030135      124      111      115
      030140      105      040      117
      030143      125      124      056
      030146      000
3310                                     .even
3311
```



3313  
 3314  
 3315  
 3316  
 3317  
 3318  
 3319  
 3320  
 3321  
 3322  
 3323  
 3324  
 3325  
 3326  
 3327  
 3328  
 3329  
 3330  
 3331  
 3332 030150  
 3333  
 3334 030150 010046  
 3335 030152 010146  
 3336 030154 010446  
 3337 030156 012737 002100 016602'  
 3338 030164 004737 033416'  
 3339 030170 017704 150030 10\$:  
 3340 030174 032704 000200  
 3341 030200 001004  
 3342 030202  
 030202 104422  
 3343 030204 005737 016602'  
 3344 030210 001367  
 3345 030212 20\$:  
 3346

```

*****
:
: SUBROUTINE - CKINTR
:
: THIS SUBROUTINE WILL INITIATE A WAIT LOOP, AND
: LOOK FOR THE INTERRUPT SUMMARY BIT TO SET. AS
: THIS ROUTINE IS USED EXCLUSIVELY TO WAIT FOR
: EITHER DNI, FATL, OR USCI BITS TO SET. ANY OTHER
: INTERRUPT BITS SETTING PRIOR TO THESE WILL BE
: FLAGGED AS UNEXPECTED INTERRUPTS.
:
: INPUTS: NONE
:
: OUTPUTS:
:
: CALLING SEQUENCE: JSR PC,CKINTR ; WAIT FOR INTERRUPT
:
*****
    
```

```

CKINTR:
    MOV R0,-(SP) ; SAVE R0
    MOV R1,-(SP) ; SAVE R1
    MOV R4,-(SP) ; SAVE R4
    MOV #2100,METER ; PUT ENOUGH TIME ON TIMER
    JSR PC,TIMON ; TURN ON CLOCK
    10$: MOV @PCSR0,R4 ; READ AND SAVE PCSRO DATA
    BIT #INTR,R4 ; INTERRUPT OCCURRED?
    BNE 20$ ; YES, DECODE IT
    BREAK ; NO, VISIT DRS
    TST METER TRAP C$BRK ; HAS TIMER EXPIRED?
    BNE 10$ ; NO, LOOK AGAIN
    20$:
    
```

```
3348 ;*****
3349 ;
3350 ; SUBROUTINE - CLRBUF
3351 ;
3352 ; THIS SUBROUTINE WILL CLEAR BOTH THE SOFTWARE BUFFERS NAMED
3353 ; RECEIVE BUFFER (RBUF) AND TRANSMIT BUFFER (TBUF), BY CALLING
3354 ; IN SEQUENCE, SUBROUTINES 'CLRCV' AND 'CLRXT'.
3355 ;
3356 ; INPUT: NONE
3357 ;
3358 ; OUTPUT: NONE
3359 ;
3360 ; SUBSIDIARY ROUTINES: SUBROUTINES 'CLRCV' AND 'CLRXT'
3361 ;
3362 ; PARAMTERS MODIFIED: ON EXIT BOTH RBUF AND TBUF WILL BE CLEARED
3363 ;
3364 ; CALL: JSR PC,CLRBUF
3365 ;
3366 ;*****
3367
3368 030212 CLRBUF:
3369 030212 004737 030236' JSR PC,CLRCV ;CLEAR RECEIVE BUFFERS
3370 030216 004737 030614' JSR PC,CLRXT ;CLEAR TRANSMIT BUFFERS
3371 030222 000207 RTS PC
3372
```

3374  
3375  
3376  
3377  
3378  
3379  
3380  
3381  
3382  
3383  
3384  
3385  
3386  
3387  
3388  
3389  
3390  
3391  
3392 030224  
3393  
3394 030224  
3395 030224 005304  
3396 030226 002402  
3397 030230 105023  
3398 030232 000774  
3399 030234  
3400 030234 000207  
3401

```
*****  
: SUBROUTINE - CLBYTE  
: THIS ROUTINE WILL CLEAR A NUMBER OF BYTES (NUMBER PASSED IN R4),  
: STARTING AT ADDRESS POINTED TO BY R3.  
: INPUT - R3 POINTS TO STARTING ADDRESS OF BYTES  
: R4 CONTAINS NUMBER OF BYTES TO BE CLEARED  
: OUTPUT - NONE  
: CALL: MOV  X,R3 ;STARTING ADDRESS OF BYTES  
: MOV  Y,R4 ;NUMBER OF BYTES TO BE CLEARED  
: JSR  PC,CLBYTE ;CLEAR THE BYTES  
*****  
CLBYTE:  
1$: DEC R4  
BLT 2$  
CLRB (R3)+  
BR 1$  
2$: RTS PC
```

```
3403 :*****  
3404 :  
3405 : SUBROUTINE CLRCV  
3406 :  
3407 : THIS SUBROUTINE WILL CLEAR ALL LOCATIONS IN SOFTWARE BUFFER  
3408 : 'RBUF'.  
3409 :  
3410 : INPUT: NONE  
3411 :  
3412 : OUTPUT: NONE  
3413 :  
3414 : SUBSIDIARY ROUTINES: SUBROUTINE 'CLBYTE'  
3415 :  
3416 : CALL: JSR PC,CLRCV  
3417 :  
3418 :*****  
3419 :  
3420 030236 CLRCV:  
3421 030236 010346 MOV R3,-(SP)  
3422 030240 010446 MOV R4,-(SP)  
3423 030242 012703 006440' MOV #RBUF,R3  
3424 030246 013704 006436' MOV RBUF-2,R4  
3425 030252 004737 030224' JSR PC,CLBYTE  
3426 030256 012604 MOV (SP)+,R4  
3427 030260 012603 MOV (SP)+,R3  
3428 030262 000207 RTS PC
```

```

3430
3431 :*****
3432 :
3433 :   SUBROUTINE - CLRDNI
3434 :
3435 :   THIS SUBROUTINE PERFORMS A WRITE ONE TO CLEAR OPERATION ON
3436 :   THE DNI BIT AND VERIFIES ITS SUCCESS.
3437 :
3438 :   INPUTS: NONE
3439 :
3440 :   OUTPUTS:      IF SUCCESSFUL ( DNI = 0 )
3441 :                  THEN CARRY = 0
3442 :
3443 :                  IF UNSUCCESSFUL ( DNI = 1 )
3444 :                      THEN CARRY = 1
3445 :                      PCSRO -> EPCSRO
3446 :                      PCSR1 -> EPCSR1
3447 :
3448 :   CALLING SEQUENCE:
3449 :       JSR      PC,CLRDNI
3450 :
3451 :*****
    
```

```

3452
3453 030264 CLRDNI:
3454 030264 010446      MOV      R4,-(SP)           ; SAVE R4
3455 030266 017737 147732 000236'  MOV      @PCSRO,PCSROC      ; READ AND SAVE PCSRO DATA
3456 030274 042737 173400 000236'  BIC      #173400,PCSROC     ; MASK ALL UPPER BYTE EXCEPT DNI
3457 030302 113777 000237' 147724  MOVB     PCSROC+1,@PCSROUB  ; CLEAR DNI
3458 030310 017704 147710      MOV      @PCSRO,R4         ; PCSRO -> R4
3459 030314 032704 004000      BIT      #DNI,R4         ; DNI = 0 ?
3460 030320 001407      BEQ     10$              ; YES
3461 030322 010437 016514'      MOV      R4,EPCSRO        ; NO, PCSRO -> EPCSRO
3462 030326 017737 147674 016516'  MOV      @PCSR1,EPCSR1     ; PCSR1 -> EPCSR1
3463 030334 000261      SEC                      ; SET CARRY
3464 030336 000401      BR      20$
3465 030340 000241 10$:      CLC                      ; CLEAR CARRY
3466 030342 012604 20$:      MOV      (SP)+,R4        ; RESTORE R4
3467 030344 000207      RTS      PC             ; AND RETURN
    
```

3469  
3470  
3471  
3472  
3473  
3474  
3475  
3476  
3477  
3478  
3479  
3480  
3481  
3482  
3483  
3484  
3485  
3486  
3487  
3488  
3489  
3490  
3491

```
*****  
: SUBROUTINE CLINTR  
:  
: THIS SUBROUTINE CLEARS LOWER BYTE OF PCSRO (DISABLE INTERRUPTS),  
: THEN SAVES PCSRO DATA. IT THEN WRITES UPPER BYTE OF SAVED DATA  
: TO THE UPPER BYTE OF PCSRO IN ORDER TO CLEAR ANY INTERRUPT BITS  
: (WRITE 1 TO CLEAR), THAT HAVE BEEN PREVIOUSLY SET.  
:  
: INPUTS: NONE  
:  
: OUTPUTS: NONE  
:  
: CALLING SEQUENCE: JSR PC,CLINTR  
:*****
```

```
CLINTR:  
MOV @PCSRO,PCSROC ;SAVE PCSRO DATA  
MOVB PCSROC+1,@PCSROUB ;CLEAR STATUS BITS IN PCSRO UPPER BYTE  
RTS PC
```

```

3493 :*****
3494 :
3495 :   SUBROUTINE - CLRRCE
3496 :
3497 :   THIS SUBROUTINE PERFORMS A WRITE ONE TO CLEAR OPERATION ON
3498 :   THE RCBI BIT AND VERIFIES ITS SUCCESS.
3499 :
3500 :   INPUTS: NONE
3501 :
3502 :   OUTPUTS:      IF SUCCESSFUL ( RCBI = 0 )
3503 :                  THEN CARRY = 0
3504 :
3505 :                  IF UNSUCCESSFUL ( RCBI = 1 )
3506 :                  THEN CARRY = 1
3507 :                  PCSRO -> EPCSRO
3508 :                  PCSR1 -> EPCSR1
3509 :
3510 :   CALLING SEQUENCE:
3511 :                   JSR   PC,CLRRCE
3512 :
3513 :*****
    
```

```

3514
3515 030364      CLRRCE:
3516 030364 010446      MOV   R4,-(SP)           ; SAVE R4
3517 030366 112777 000004 147640      MOVB  @RCBIB,@PCSROUB      ; WRITE ONE TO CLEAR RCBI BIT
3518 030374 017704 147624      MOV   @PCSRO,R4           ; PCSRO -> R4
3519 030400 032704 002000      BIT   @RCBI,R4           ; RCBI = 0 ?
3520 030404 001407      BEQ   10$             ; YES
3521 030406 010437 016514'      MOV   R4,EPCSRO         ; NO, PCSRO -> EPCSRO
3522 030412 017737 147610 016516'  MOV   @PCSR1,EPCSR1     ; PCSR1 -> EPCSR1
3523 030420 000261      SEC                       ; SET CARRY
3524 030422 000401      BR    20$             ;
3525 030424 000241 10$:      CLC                       ; CLEAR CARRY
3526 030426 012604 20$:      MOV   (SP)+,R4         ; RESTORE R4
3527 030430 000207      RTS    PC             ; AND RETURN
    
```

3529  
 3530  
 3531  
 3532  
 3533  
 3534  
 3535  
 3536  
 3537  
 3538  
 3539  
 3540  
 3541  
 3542  
 3543  
 3544  
 3545  
 3546  
 3547  
 3548  
 3549  
 3550  
 3551

```

*****
SUBROUTINE - CLRRXI
THIS SUBROUTINE PERFORMS A WRITE ONE TO CLEAR OPERATION ON
THE RXI BIT AND VERIFIES ITS SUCCESS.

INPUTS: NONE

OUTPUTS:      IF SUCCESSFUL ( RXI = 0 )
                THEN CARRY = 0

                IF UNSUCCESSFUL ( RXI = 1 )
                THEN CARRY = 1
                PCSRO -> EPCSRO
                PCSR1 -> EPCSR1

CALLING SEQUENCE:
                JSR      PC,CLRRXI
*****
    
```

3552 030432  
 3553 030432 010446  
 3554 030434 112777 000040 147572  
 3555 030442 017704 147556  
 3556 030446 032704 020000  
 3557 030452 001407  
 3558 030454 010437 016514'  
 3559 030460 017737 147542 016516'  
 3560 030466 000261  
 3561 030470 000401  
 3562 030472 000241  
 3563 030474 012604  
 3564 030476 000207

```

CLRRXI:
MOV      R4,-(SP)           ; SAVE R4
MOV      @RXIB,@PCSROUB    ; WRITE ONE TO CLEAR RXI BIT
MOV      @PCSRO,R4         ; PCSRO -> R4
BIT      @RXI,R4           ; RXI = 0 ?
BEQ      10$               ; YES
MOV      R4,EPCSRO         ; NO, PCSRO -> EPCSRO
MOV      @PCSR1,EPCSR1     ; PCSR1 -> EPCSR1
SEC                         ; SET CARRY
BR       20$

10$:    CLC                 ; CLEAR CARRY
20$:    MOV      (SP)+,R4   ; RESTORE R4
RTS     PC                 ; AND RETURN
    
```



3566  
 3567  
 3568  
 3569  
 3570  
 3571  
 3572  
 3573  
 3574  
 3575  
 3576  
 3577  
 3578  
 3579  
 3580  
 3581  
 3582  
 3583  
 3584  
 3585  
 3586  
 3587  
 3588  
 3589 030500  
 3590 030500 010446  
 3591 030502 112777 000020 147524  
 3592 030510 017704 147510  
 3593 030514 032704 010000  
 3594 030520 001407  
 3595 030522 010437 016514'  
 3596 030526 017737 147474 016516'  
 3597 030534 000261  
 3598 030536 000401  
 3599 030540 000241  
 3600 030542 012604  
 3601 030544 000207

```

*****
:
: SUBROUTINE - CLRTXI
:
: THIS SUBROUTINE PERFORMS A WRITE ONE TO CLEAR OPERATION ON
: THE TXI BIT AND VERIFIES ITS SUCCESS.
:
: INPUTS: NONE
:
: OUTPUTS: IF SUCCESSFUL ( TXI = 0 )
:          THEN CARRY = 0
:
:          IF UNSUCCESSFUL ( TXI = 1 )
:          THEN CARRY = 1
:          PCSRO -> EPCSRO
:          PCSR1 -> EPCSR1
:
: CALLING SEQUENCE:
:          JSR PC,CLRTXI
*****
    
```

```

CLRTXI:
MOV     R4, -(SP)           ; SAVE R4
MOVB   @TXIB, @PCSROUB     ; WRITE ONE TO CLEAR TXI BIT
MOV     @PCSRO, R4         ; PCSRO -> R4
BIT     @TXI, R4           ; TXI = 0 ?
BEQ     10$                ; YES
MOV     R4, EPCSRO         ; NO, PCSRO -> EPCSRO
MOV     @PCSR1, EPCSR1     ; PCSR1 -> EPCSR1
SEC                     ; SET CARRY
BR      20$
10$:   CLC                  ; CLEAR CARRY
20$:   MOV     (SP)+, R4     ; RESTORE R4
RTS     PC                  ; AND RETURN
    
```

```

3603 ;*****
3604 ;
3605 ; SUBROUTINE - CLRSER
3606 ;
3607 ; THIS SUBROUTINE PERFORMS A WRITE ONE TO CLEAR OPERATION ON
3608 ; THE SERI BIT AND VERIFIES ITS SUCCESS.
3609 ;
3610 ; INPUTS: NONE
3611 ;
3612 ; OUTPUTS: IF SUCCESSFUL ( SERI = 0 )
3613 ; THEN CARRY = 0
3614 ;
3615 ; IF UNSUCCESSFUL ( SERI = 1 )
3616 ; THEN CARRY = 1
3617 ; PCSRO -> EPCSRO
3618 ; PCSR1 -> EPCSR1
3619 ;
3620 ; CALLING SEQUENCE:
3621 ; JSR PC,CLRSER
3622 ;
3623 ;*****
    
```

```

3624 CLRSER:
3625 030546 MOV R4,-(SP) ; SAVE R4
3626 030546 010446 MOV #SERIB,@PCSROUB ; WRITE ONE TO CLEAR SERI BIT
3627 030550 112777 000200 147456 MOV @PCSRO,R4 ; PCSRO -> R4
3628 030556 017704 147442 BIT #SERI,R4 ; SERI = 0 ?
3629 030562 032704 100000 BEQ 10$ ; YES
3630 030566 001407 MOV R4,EPCSRO ; NO, PCSRO -> EPCSRO
3631 030570 010437 016514' MOV @PCSR1,EPCSR1 ; PCSR1 -> EPCSR1
3632 030574 017737 147426 016516' SEC ; SET CARRY
3633 030602 000261 BR 20$
3634 030604 000401 10$: CLC ; CLEAR CARRY
3635 030606 000241 20$: MOV (SP)+,R4 ; RESTORE R4
3636 030610 012604 RTS ; AND RETURN
3637 030612 000207 PC
    
```

3639  
3640  
3641  
3642  
3643  
3644  
3645  
3646  
3647  
3648  
3649  
3650  
3651  
3652  
3653  
3654  
3655  
3656  
3657 030614  
3658 030614 010346  
3659 030616 010446  
3660 030620 012703 002436'  
3661 030624 013704 002434'  
3662 030630 004737 030224'  
3663 030634 012604  
3664 030636 012603  
3665 030640 000207  
3666

```
*****  
: SUBROUTINE CLRXMT  
: THIS SUBROUTINE WILL CLEAR ALL LOCATIONS IN SOFTWARE BUFFER  
: TBUF'.  
: INPUT: NONE  
: OUTPUT: NONE  
: SUBSIDIARY ROUTINES: SUBROUTINE CLBYTE  
: CALL JSR PC,CLRXMT  
*****  
CLRXMT:  
MOV R3,-(SP)  
MOV R4,-(SP)  
MOV @TBUF,R3  
MOV TBUF-2,R4  
JSR PC,CLBYTE  
MOV (SP)+,R4  
MOV (SP)+,R3  
RTS PC
```

3668  
 3669  
 3670  
 3671  
 3672  
 3673  
 3674  
 3675  
 3676  
 3677  
 3678  
 3679  
 3680  
 3681  
 3682  
 3683  
 3684  
 3685  
 3686  
 3687  
 3688  
 3689  
 3690  
 3691  
 3692  
 3693  
 3694 030642  
 3695 030642 010346  
 3696 030644 010446  
 3697 030646 012703 016574'  
 3698 030652 010504  
 3699 030654 022324  
 3700 030656 001004  
 3701 030660 022324  
 3702 030662 001002  
 3703 030664 000241  
 3704 030666 000406  
 3705 030670 012703 016570'  
 3706 030674 010504  
 3707 030676 012423  
 3708 030700 012423  
 3709 030702 000261  
 3710 030704 012604  
 3711 030706 012603  
 3712 030710 000207

```

*****
:
:   SUBROUTINE - CMPCRC
:
:   THIS SUBROUTINE COMPARES A CRC VALUE WITH
:   AN EXPECTED CRC VALUE.
:
:   INPUTS:           R5 = ADDRESS OF ACTUAL CRC VALUE RECEIVED.
:
:   INPLICIT INPUTS:  XCRC = EXPECTED CRC VALUE
:
:   OUTPUTS:          IF SUCCESSFUL CRC COMPARE
:                       THEN CARRY = 0
:
:                       IF UNSUCCESSFUL CRC COMPARE
:                           THEN CARRY = 1
:                           EXPECTED CRC = XCRC
:                           ACTUAL CRC  -> ECRC
:
:   CALLING SEQUENCE:
:   JSR   PC,CMPCRC
:
*****
    
```

```

CMPCRC:
    MOV   R3, -(SP)           ; SAVE R3
    MOV   R4, -(SP)           ; SAVE R4
    MOV   @XCRC, R3           ; R3 POINTS TO EXPECTED CRC
    MOV   R5, R4              ; R4 POINTS TO ACTUAL CRC
    CMP   (R3)+, (R4)+        ; FIRST CRC WORD COMPARE ?
    BNE   10$                 ; NO
    CMP   (R3)+, (R4)+        ; SECOND CRC WORD COMPARE ?
    BNE   10$                 ; NO
    CLC                          ; YES, CLEAR CARRY
    BR   20$
10$:   MOV   @ECRC, R3         ; POINT TO ERROR TABLE
    MOV   R5, R4              ; POINT TO ACTUAL DATA
    MOV   (R4)+, (R3)+        ; LOAD ECRC TABLE
    MOV   (R4)+, (R3)+
    SEC                          ; AND SET CARRY
20$:   MOV   (SP)+, R4         ; RESTORE R4
    MOV   (SP)+, R3         ; RESTORE R3
    RTS   PC                  ; AND RETURN
    
```

3714  
 3715  
 3716  
 3717  
 3718  
 3719  
 3720  
 3721  
 3722  
 3723  
 3724  
 3725  
 3726  
 3727  
 3728  
 3729  
 3730  
 3731  
 3732  
 3733  
 3734  
 3735  
 3736 030712  
 3737 030712  
 3738 030714  
 3739 030716  
 3740 030720  
 3741 030722  
 3742 030726  
 3743 030732  
 3744 030734  
 3745 030736  
 3746 030740  
 3747 030742  
 3748 030744  
 3749 030750  
 3750 030754  
 3751 030756  
 3752 030760  
 3753 030762  
 3754 030764  
 3755 030766  
 3756 030770  
 3757

```

*****
:
:      SUBROUTINE - CMPDAT
:
:      THIS SUBROUTINE COMPARES THE RECEIVE BUFFER (RBUF) DATA FIELD
:      WITH THE TRANSMIT BUFFER (TBUF) DATA FIELD.
:
:      INPUTS:          R5 = NUMBER OF WORDS TO COMPARE
:
:      OUTPUTS:        IF SUCCESSFUL DATA COMPARE
:                      THEN CARRY = 0
:
:                      IF UNSUCCESSFUL DATA COMPARE
:                      THEN CARRY = 1
:                      EXPECTED DATA -> XDAT
:                      ACTUAL DATA  -> EDAT
:
:      CALLING SEQUENCE:
:                      JSR      PC,CMPDAT
:
*****
    
```

```

CMPDAT:
      MOV      R0,-(SP)          ; SAVE R0
      MOV      R3,-(SP)          ; SAVE R3
      MOV      R4,-(SP)          ; SAVE R4
      MOV      R5,R0             ; R0 = NUMBER OF WORDS TO COMPARE
      MOV      @TBUF+14.,R3      ; R3 POINTS TO EXPECTED DATA
      MOV      @RBUF+14.,R4      ; R4 POINTS TO ACTUAL DATA
10$:   CMP      (R3)+,(R4)+      ; DATA COMPARE ?
      BNE      20$              ; NO
      DEC      R0                ; YES, DONE ?
      BNE      10$              ; NO
      BR       30$              ; YES
20$:   MOV      -(R3),XDAT       ; SAVE EXPECTED DATA
      MOV      -(R4),EDAT       ; SAVE ACTUAL ERROR DATA
      SEC                          ; SET CARRY
      BR       40$
30$:   CLC                          ; CLEAR CARRY
40$:   MOV      (SP)+,R4         ; RESTORE R4
      MOV      (SP)+,R3         ; RESTORE R3
      MOV      (SP)+,R0         ; RESTORE R0
      RTS      PC                ; AND RETURN
    
```

```

3759
3760 ;*****
3761 ;
3762 ; SUBROUTINE - CMPMEM
3763 ;
3764 ; THIS SUBROUTINE COMPARES THE READ MEMORY BUFFER (RBUF)
3765 ; WITH THE WRITE MEMORY BUFFER (TBUF).
3766 ;
3767 ; INPUTS: R5 = NUMBER OF WORDS TO COMPARE
3768 ;
3769 ; OUTPUTS: IF SUCCESSFUL DATA COMPARE
3770 ; THEN CARRY = 0
3771 ;
3772 ; IF UNSUCCESSFUL DATA COMPARE
3773 ; THEN CARRY = 1
3774 ; EXPECTED DATA -> XDAT
3775 ; ACTUAL DATA -> EDAT
3776 ;
3777 ; CALLING SEQUENCE:
3778 ; JSR PC,CMPMEM
3779 ;
3780 ;*****
    
```

```

3781
3782 030772 CMPMEM:
3783 030772 010046 MOV R0,-(SP) ; SAVE R0
3784 030774 010346 MOV R3,-(SP) ; SAVE R3
3785 030776 010446 MOV R4,-(SP) ; SAVE R4
3786 031000 010500 MOV R5,R0 ; R0 = NUMBER OF WORDS TO COMPARE
3787 031002 012703 002436' MOV #TBUF,R3 ; R3 POINTS TO EXPECTED DATA
3788 031006 012704 006440' MOV #RBUF,R4 ; R4 POINTS TO ACTUAL DATA
3789 031012 022324 10$: CMP (R3)+,(R4)+ ; DATA COMPARE ?
3790 031014 001003 BNE 20$ ; NO
3791 031016 005300 DEC R0 ; YES, DONE ?
3792 031020 001374 BNE 10$ ; NO
3793 031022 000406 BR 30$ ; YES
3794 031024 014337 016566' 20$: MOV -(R3),XDAT ; SAVE EXPECTED DATA
3795 031030 014437 016564' MOV -(R4),EDAT ; SAVE ACTUAL ERROR DATA
3796 031034 000261 SEC ; SET CARRY
3797 031036 000401 BR 40$
3798 031040 000241 30$: CLC ; CLEAR CARRY
3799 031042 012604 40$: MOV (SP)+,R4 ; RESTORE R4
3800 031044 012603 MOV (SP)+,R3 ; RESTORE R3
3801 031046 012600 MOV (SP)+,R0 ; RESTORE R0
3802 031050 000207 RTS PC ; AND RETURN
    
```

3804  
 3805  
 3806  
 3807  
 3808  
 3809  
 3810  
 3811  
 3812  
 3813  
 3814  
 3815  
 3816  
 3817  
 3818  
 3819  
 3820  
 3821  
 3822  
 3823  
 3824  
 3825  
 3826  
 3827 031052  
 3828 031052 010046  
 3829 031054 010346  
 3830 031056 010446  
 3831 031060 010500  
 3832 031062 012703 000000  
 3833 031066 012704 006532'  
 3834 031072 020324  
 3835 031074 001003  
 3836 031076 005300  
 3837 031100 001374  
 3838 031102 000406  
 3839 031104 010337 016566'  
 3840 031110 014437 016564'  
 3841 031114 000261  
 3842 031116 000401  
 3843 031120 000241  
 3844 031122 012604  
 3845 031124 012603  
 3846 031126 012600  
 3847 031130 000207

```

*****
:
: SUBROUTINE - CMPRNT
:
: THIS SUBROUTINE COMPARES THE RECEIVE BUFFER (RBUF) DATA FIELD
: TO VERIFY ZERO PADDING HAS OCCURED.
:
: INPUTS: R5 = NUMBER OF WORDS TO COMPARE
:
: OUTPUTS: IF SUCCESSFUL DATA COMPARE
: THEN CARRY = 0
:
: IF UNSUCCESSFUL DATA COMPARE
: THEN CARRY = 1
: EXPECTED DATA -> XDAT
: ACTUAL DATA -> EDAT
:
: CALLING SEQUENCE:
: JSR PC,CMPRNT
*****
    
```

```

CMPRNT:
MOV R0,-(SP) ; SAVE R0
MOV R3,-(SP) ; SAVE R3
MOV R4,-(SP) ; SAVE R4
MOV R5,R0 ; R0 = NUMBER OF WORDS TO COMPARE
MOV @ZERO,R3 ; R3 IS EXPECTED DATA (ZERO'S)
MOV @RBUF+58.,R4 ; R4 POINTS TO ACTUAL DATA
10$: CMP R3,(R4)+ ; DATA = ZERO'S ?
BNE 20$ ; NO
DEC R0 ; YES, DONE ?
BNE 10$ ; NO
BR 30$ ; YES
20$: MOV R3,XDAT ; SAVE EXPECTED DATA
MOV -(R4),EDAT ; SAVE ACTUAL ERROR DATA
SEC ; SET CARRY
BR 40$
30$: CLC ; CLEAR CARRY
40$: MOV (SP)+,R4 ; RESTORE R4
MOV (SP)+,R3 ; RESTORE R3
MOV (SP)+,R0 ; RESTORE R0
RTS PC ; AND RETURN
    
```

3849  
 3850  
 3851  
 3852  
 3853  
 3854  
 3855  
 3856  
 3857  
 3858  
 3859  
 3860  
 3861  
 3862  
 3863  
 3864  
 3865  
 3866 031132  
 3867 031132 010437 016602'  
 3868 031136 004737 033416'  
 3869 031142  
       031142 104422  
 3870 031144 005737 016602'  
 3871 031150 001374  
 3872 031152 004737 033402'  
 3873 031156 000207  
 3874  
 3875

```

*****
:
:      SUBROUTINE - DELAY
:
:      THIS SUBROUTINE WILL USE THE SYTEM CLOCK TO ENABLE A WAITING
:      PERIOD DETERMINED BY THE VALUE PASSED IN R4.
:
:      INPUT: R4 CONTAINS DELAY VALUE
:
:      OUTPUT: NONE
:
:      CALL:  MOV    #DELAY_VALUE,R4
:            JSR    PC,DELAY
:
*****
    
```

```

DELAY:
      MOV    R4,METER          ;GET DELAY VALUE
      JSR    PC,TIMON         ;START DELAY
1$:   BREAK                    ;VISIT DRS WHILE WAITING
                                TRAP    C$BRK
      TST    METER            ;FINISHED?
      BNE    1$               ;CONTINUE WAIT
      JSR    PC,TIMOFF        ;TURN OFF SYSTEM CLOCK
      RTS    PC               ;RETURN TO CALLING ROUTINE
    
```



3877  
 3878  
 3879  
 3880  
 3881  
 3882  
 3883  
 3884  
 3885  
 3886  
 3887  
 3888  
 3889  
 3890  
 3891  
 3892  
 3893  
 3894  
 3895  
 3896  
 3897 031160  
 3898 031160 010146  
 3899 031162 010246  
 3900 031164 010546  
 3901 031166 042705 177400  
 3902 031172 074504  
 3903 031174 013701 016614'  
 3904 031200 013702 016616'  
 3905 031204 012705 000010  
 3906 031210  
 3907 031210 000241  
 3908 031212 006003  
 3909 031214 006004  
 3910 031216 103002  
 3911 031220 074103  
 3912 031222 074204  
 3913 031224  
 3914 031224 077507  
 3915 031226 012605  
 3916 031230 012602  
 3917 031232 012601  
 3918 031234 000207  
 3919

```

*****
SUBROUTINE - GETCRC
THIS SUBROUTINE IS A BYTE WISE 32-BIT CRC CALCULATOR
INPUTS: R5 CONTAINS NEW BYTE TO ADD TO CRC
        R3,R4 CONTAIN CURRENT PARTIAL CRC CODE
IMPLICIT INPUTS:      POLYH = CRC FUNCTION POLYNOMIAL HIGH WORD
                     POLYL = CRC FUNCTION POLYNOMIAL LOW WORD
OUTPUTS:              R3,R4 CONTAIN UPDATED CRC
CALLING SEQUENCE:    MOVB (R1)+,R5      ;GET NEXT BYTE
                     JSR   PC,GETCRC   ;CALCULATE CRC
*****
    
```

```

GETCRC:
MOV     R1,-(SP)      ;SAVE R1
MOV     R2,-(SP)      ;SAVE R2
MOV     R5,-(SP)      ;SAVE R5
BIC     #1C377,R5     ;CLEAR HIGH BYTE
XOR     R5,R4         ;MERGE NEW BYTE WITH OLD CRC
MOV     POLYH,R1      ;GET CRC POLYNOMIAL HIGH WORD
MOV     POLYL,R2      ;GET CRC POLYNOMIAL LOW WORD
MOV     #8.,R5        ;LOOP COUNT
1$:
CLC                     ;CLEAR THE CARRY
ROR     R3             ;SHIFT RIGHT THE CRC
ROR     R4             ;32 BITS WORTH
BCC     2$            ;SKIP IF BIT 0 NOT SET
XOR     R1,R3         ;EXCLUSIVE OR IN THE POLY
XOR     R2,R4         ;BOTH HIGH AND LOW WORDS
2$:
SOB     R5,1$         ;AND LOOP ON ALL 8 BITS
MOV     (SP)+,R5      ;RESTORE R5
MOV     (SP)+,R2      ;RESTORE R2
MOV     (SP)+,R1      ;RESTORE R1
RTS     PC            ;RETURN TO CALLING PROGRAM
    
```

```

3921
3922
3923
3924
3925
3926
3927
3928
3929
3930
3931
3932
3933
3934
3935
3936
3937
3938
3939
3940
3941 031236
3942 031236 010046
3943 031240 010346
3944 031242 010546
3945
3946 031244 012700 000006
3947 031250 012703 102247'
3948 031254 012705 102154'
3949
3950 031260 112537 102202'
3951 031264 004737 031320'
3952 031270 113723 102203'
3953 031274 004737 031356'
3954 031300 113723 102203'
3955 031304 105723
3956 031306 077014
3957
3958 031310 012605
3959 031312 012603
3960 031314 012600
3961 031316 000207
    
```

```

;*****
;
;      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
;*****
HEXDPA:
    MOV    R0,-(SP)      ; SAVE R0
    MOV    R3,-(SP)      ; SAVE R3
    MOV    R5,-(SP)      ; SAVE R5
;
    MOV    #6,R0         ; DO LOOP = 6 BYTES
    MOV    #DEFADR,R3    ; POINT TO ASCII MESSAGE
    MOV    #DPA,R5       ; POINT TO DEFAULT PHYSICAL ADDR
;
10$:   MOVB  (R5)+,HEXDAT ; LOAD BYTE FOR CONVERSION
       JSR  PC,HEXH      ; CONVERT HIGH NIBBLE
       MOVB HEXVAL,(R3)+ ; LOAD INTO ASCII MESSAGE
       JSR  PC,HEXL      ; CONVERT LOW NIBBLE
       MOVB HEXVAL,(R3)+ ; LOAD INTO ASCII MESSAGE
       TSTB (R3)+       ; SKIP OVER HYPHEN IN MESSAGE
       SOB  R0,10$      ; LOOP TILL ALL 6 BYTES ARE DONE
;
    MOV    (SP)+,R5      ; RESTORE R5
    MOV    (SP)+,R3      ; RESTORE R3
    MOV    (SP)+,R0      ; RESTORE R0
    RTS    PC           ; AND RETURN
    
```

```

3963
3964 :*****
3965 :
3966 :   SUBROUTINE - HEXH
3967 :
3968 :   THIS SUBROUTINE LOADS HEXVAL WITH THE ASCII HEX VALUE
3969 :   FOR THE HIGH NIBBLE IN HEXDAT
3970 :
3971 :   INPUTS:          NONE
3972 :
3973 :   IMPLICIT
3974 :   INPUTS:          HEXDAT = BYTE TO BE CONVERTED
3975 :
3976 :   OUTPUTS:         HEXVAL = ASCII HEX VALUE FOR THE HIGH NIBBLE
3977 :
3978 :   CALLING SEQUENCE:
3979 :                   JSR     PC,HEXH
3980 :
3981 :*****
    
```

```

3982
3983 031320
3984 031320 010146
3985
3986 031322 013701 102202'
3987 031326 042701 177417
3988
3989 031332 006201
3990 031334 006201
3991 031336 006201
3992 031340 006201
3993
3994 031342 062701 102273'
3995 031346 111137 102203'
3996
3997 031352 012601
3998 031354 000207
    
```

```

HEXH:
MOV     R1,-(SP)          ; SAVE R1
:
MOV     HEXDAT,R1        ; LOAD DATA FOR CONVERSION
BIC     @177417,R1       ; MASK HIGH NIBBLE
:
ASR     R1                ; SHIFT RIGHT
ASR     R1
ASR     R1
ASR     R1
:
ADD     @HEXTBL,R1        ; GET INDEX INTO HEXTBL
MOVB   (R1),HEXVAL      ; AND LOAD HEXVAL
:
MOV     (SP)+,R1         ; RESTORE R1
RTS    PC                ; AND RETURN
    
```

```
4000  
4001 ;*****  
4002 ;  
4003 ; SUBROUTINE - HEXL  
4004 ;  
4005 ; THIS SUBROUTINE LOADS HEXVAL WITH THE ASCII HEX VALUE  
4006 ; FOR THE LOW NIBBLE IN HEXDAT  
4007 ;  
4008 ; INPUTS: NONE  
4009 ;  
4010 ; IMPLICIT  
4011 ; INPUTS: HEXDAT = BYTE TO BE CONVERTED  
4012 ;  
4013 ; OUTPUTS: HEXVAL = ASCII HEX VALUE FOR THE LOW NIBBLE  
4014 ;  
4015 ; CALLING SEQUENCE:  
4016 ; JSR PC,HEXL  
4017 ;  
4018 ;*****  
4019  
4020 031356 HEXL: MOV R1,-(SP) ; SAVE R1  
4021 031356 010146 ;  
4022 ; MOV HEXDAT,R1 ; LOAD DATA FOR CONVERSION  
4023 031360 013701 102202' BIC #177760,R1 ; MASK LOW NIBBLE  
4024 031364 042701 177760 ;  
4025 ; ADD #HEXTBL,R1 ; GET INDEX INTO HEXTBL  
4026 031370 062701 102273' MOV (R1),HEXVAL ; AND LOAD HEXVAL  
4027 031374 111137 102203' ;  
4028 ;  
4029 031400 012601 MOV (SP)+,R1 ; RESTORE R1  
4030 031402 000207 RTS PC ; AND RETURN
```

```

4032
4033
4034
4035
4036
4037
4038
4039
4040
4041
4042
4043
4044
4045
4046
4047
4048
    ;*****
    ;
    ; SUBROUTINE - LDBUF
    ;
    ; THIS SUBROUTINE LOADS TBUF WITH AN ADDRESS DATA PATTERN
    ; STARTING WITH THE ADDRESS POINTED TO BY R5
    ;
    ; INPUTS:          R5 = ADDRESS OF SPECIFIED DATA ADDRESS
    ;
    ; OUTPUTS:         TBUF = ADDRESS DATA PATTERN
    ;
    ; CALLING SEQUENCE:
    ;                   JSR    PC,LDBUF
    ;*****
    
```

```

4049 031404
4050 031404 010146
4051 031406 010346
4052 031410 010446
4053 031412 012701 002000
4054 031416 011504
4055 031420 012703 002436'
4056 031424 010423
4057 031426 062704 000002
4058 031432 005301
4059 031434 001373
4060 031436 012604
4061 031440 012603
4062 031442 012601
4063 031444 000207

LDBUF:
    MOV    R1,-(SP)          ; SAVE R1
    MOV    R3,-(SP)          ; SAVE R3
    MOV    R4,-(SP)          ; SAVE R4
    MOV    #1024.,R1         ; DO 1024. WORDS
    MOV    (R5),R4           ; R4 = STARTING DATA ADDRESS
    MOV    #TBUF,R3         ; R3 POINTS TO TBUF
10$:    MOV    R4,(R3)+       ; LOAD TBUF
    ADD    #2,R4             ; ADD 2 TO DATA
    DEC    R1                ; DONE 1K BLOCK ?
    BNE    10$              ; NO
    MOV    (SP)+,R4         ; RESTORE R4
    MOV    (SP)+,R3         ; RESTORE R3
    MOV    (SP)+,R1         ; RESTORE R1
    RTS    PC               ; AND RETURN
    
```

```

4065
4066 ;*****
4067 ;
4068 ; SUBROUTINE - LDBUFC
4069 ;
4070 ; THIS SUBROUTINE LOADS TBUF WITH THE COMPLIMENT OF AN
4071 ; ADDRESS DATA PATTERN STARTING WITH THE ADDRESS SPECIFIED BY R5
4072 ;
4073 ; INPUTS: R5 = ADDRESS OF SPECIFIED DATA ADDRESS
4074 ;
4075 ; OUTPUTS: TBUF = COMPLIMENTED ADDRESS DATA PATTERN
4076 ;
4077 ; CALLING SEQUENCE:
4078 ; JSR PC,LDBUFC
4079 ;
4080 ;*****
    
```

```

4081
4082 031446 LDBUFC:
4083 031446 010146 MOV R1,-(SP) ; SAVE R1
4084 031450 010246 MOV R2,-(SP) ; SAVE R2
4085 031452 010346 MOV R3,-(SP) ; SAVE R3
4086 031454 010446 MOV R4,-(SP) ; SAVE R4
4087 031456 012701 002000 MOV #1024.,R1 ; DO 1024. WORDS
4088 031462 011504 MOV (R5),R4 ; R4 = STARTING DATA ADDRESS
4089 031464 012703 002436' MOV #TBUF,R3 ; R3 POINTS TO TBUF
4090 031470 010402 10$: MOV R4,R2
4091 031472 005102 COM R2 ; COMPLIMENT DATA
4092 031474 010223 MOV R2,(R3)+ ; LOAD TBUF
4093 031476 062704 000002 ADD #2,R4 ; ADD 2 TO DATA
4094 031502 005301 DEC R1 ; DONE 1K BLOCK?
4095 031504 001371 BNE 10$ ; NO
4096 031506 012604 MOV (SP)+,R4 ; RESTORE R4
4097 031510 012603 MOV (SP)+,R3 ; RESTORE R3
4098 031512 012602 MOV (SP)+,R2 ; RESTORE R2
4099 031514 012601 MOV (SP)+,R1 ; RESTORE R1
4100 031516 000207 RTS PC ; AND RETURN
    
```

```

4102 ;*****
4103 ;
4104 ; SUBROUTINE - LDBUFR (USED IN ADDRESS TESTS)
4105 ;
4106 ; THIS ROUTINE SETS UP THE TRANSMIT AND RECEIVE BUFFERS. THE
4107 ; NUMBER OF BYTES IN A PACKET IS DETERMINED BY THE CONTENTS
4108 ; PASSED IN 'BYTCNT'. IN ANY CASE, A PACKET LENGHT WILL BE NO
4109 ; MORE THAN 32 BYTES, OR 36 INCLUDING CRC IF 'DOCRC' SET.
4110 ;
4111 ; A. LOAD TRANSMIT BUFFER TBUF (DEST. ADDR,SOURCE ADDR,TYPE,DATA)
4112 ; B. APPEND CRC IF 'DOCRC' FLAG SET
4113 ;
4114 ; INPUTS: DOCRC = 0 THEN NO CRC
4115 ; 1 THEN CALCULATE CRC AND APPEND
4116 ; BYTCNT= # OF DATA BYTES IN PACKET
4117 ;
4118 ; IMPLICIT INPUTS:
4119 ; DEST: = DESTINATION ADDRESS
4120 ;
4121 ; OUTPUTS:
4122 ; TBUF IS SET UP FOR TRANSMIT
4123 ;
4124 ; PARAMETERS MODIFIED: 'DOCRC' WILL ALWAYS BE CLEARED ON EXIT
4125 ;
4126 ; CALLING SEQUENCE:
4127 ; INSURE 'DOCRC' FLAG CONTAINS CORRECT DATA
4128 ;
4129 ; MOV #X,R3 ;CRC LOW WORD
4130 ; MOV #Y,R4 ;CRC HIGH WORD
4131 ; MOV #Z,BYTCNT ;NUMBER OF BYTES THIS PACKET
4132 ; JSR PC,LDBUFR
4133 ;
4134 ;*****
4135 ;
4136 031520 LDBUFR:
4137 031520 010046 MOV R0,-(SP) ; SAVE R0
4138 031522 010146 MOV R1,-(SP) ; SAVE R1
4139 031524 010246 MOV R2,-(SP) ; SAVE R2
4140 031526 010346 MOV R3,-(SP) ; SAVE R3
4141 031530 010446 MOV R4,-(SP) ; SAVE R4
4142 031532 ^10546 MOV R5,-(SP) ; SAVE R5
4143 ;
4144 ;SET UP TRANSMIT BUFFER TBUF
4145 ;LOAD DESTINATION ADDRESS
4146 ;
4147 031534 012705 000256' MOV #DEST,R5 ; POINT TO DESTINATION ADDRESS
4148 031540 012701 002436' MOV #TBUF,R1 ; POINT TO TBUF
4149 031544 012521 MOV (R5)+,(R1)+ ; LOAD DESTINATION ADDRESS
4150 031546 012521 MOV (R5)+,(R1)+
4151 031550 011521 MOV (R5),(R1)+
4152 ;
4153 ;LOAD SOURCE ADDRESS
4154 ;
4155 031552 012705 000264' MOV #SRC,R5 ; LOAD FOR LATER COMPARISON
4156 031556 012521 MOV (R5)+,(R1)+
4157 031560 012521 MOV (R5)+,(R1)+
4158 031562 011521 MOV (R5),(R1)+
    
```

```

4159
4160 ;SET TYPE FIELD
4161
4162 031564 012721 000005          MOV    #5,(R1)+          ; ENTER DIAGNOSTIC ID IN TYPE FIELD
4163
4164 ;LOAD DATA FIELD (LENGTH DEPENDENT ON CONTENTS OF 'BYTCNT')
4165
4166 031570 013700 016560          MOV    BYTCNT,R0          ; BYTE COUNT
4167 031574 012705 016626          MOV    @PATRN1,R5         ; POINT TO DATA PATTERN
4168 031600 012521          20$:  MOV    (R5)+,(R1)+      ; LOAD DATA PATTERN
4169 031602 005300          DEC    R0                 ; DONE ?
4170 031604 002375          BGE    20$                ; NO
4171
4172 ;CALCULATE CRC AND SAVE IN 'XCRC'
4173
4174 031606 010146          MOV    R1,-(SP)           ; SAVE R1
4175 031610 010246          MOV    R2,-(SP)           ; SAVE R2
4176 031612 013702 016560          MOV    BYTCNT,R2         ; GET DATA BYTE COUNT
4177 031616 006302          ASL    R2                 ; ALIGN
4178 031620 062702 000020          ADD    #16.,R2           ; ADD HEADER
4179 031624 012701 002436          MOV    @TBUF,R1          ; BASE ADDR OF TRANSMIT BUFFER
4180 031630 012703 177777          MOV    #-1,R3            ; INIT CRC
4181 031634 012704 177777          MOV    #-1,R4            ; INIT CRC
4182 031640 004737 026506          JSR    PC,BLKCRC         ; CALCULATE AND SAVE CRC
4183 031644 012602          MOV    (SP)+,R2          ; RESTORE R2
4184 031646 012601          MOV    (SP)+,R1          ; RESTORE R1
4185
4186 ;IF CRC FLAG SET, APPEND CRC, THEN SAVE EXPECTED CRC IN 'XCRC' TABLE
4187
4188 031650 005737 016562          TST    DDCRC             ;APPEND CRC?
4189 031654 001402          BEQ    30$               ;NO, SKIP APPENDING CRC
4190 031656 010421          MOV    R4,(R1)+          ;APPEND CRC LOW WORD
4191 031660 010321          MOV    R3,(R1)+          ;APPEND CRC HIGH WORD
4192
4193
4194 031662          30$:  CLR    DDCRC             ;INSURE CRC FLAG IS CLEARED
4195 031662 005037 016562'
4196
4197 031666 012605          MOV    (SP)+,R5          ; RESTORE R5
4198 031670 012604          MOV    (SP)+,R4          ; RESTORE R4
4199 031672 012603          MOV    (SP)+,R3          ; RESTORE R3
4200 031674 012602          MOV    (SP)+,R2          ; RESTORE R2
4201 031676 012601          MOV    (SP)+,R1          ; RESTORE R1
4202 031700 012600          MOV    (SP)+,R0          ; RESTORE R0
4203 031702 000207          RTS    PC                 ; AND RESTORE
4204
    
```



```

4206
4207 ;*****
4208 ;
4209 ; SUBROUTINE - LDDEST
4210 ;
4211 ; THIS SUBROUTINE LOADS A SPECIFIED DESTINATION ADDRESS
4212 ; INTO DEST: .
4213 ;
4214 ; INPUTS: R5 = ADDRESS OF SPECIFIED DESTINATION ADDRESS
4215 ;
4216 ; OUTPUTS: DEST = SPECIFIED DESTINATION ADDRESS
4217 ;
4218 ; CALLING SEQUENCE:
4219 ; JSR PC,LDDEST
4220 ;
4221 ;*****
    
```

```

4222
4223 031704 LDDEST:
4224 031704 010346 MOV R3,-(SP) ; SAVE R3
4225 031706 010446 MOV R4,-(SP) ; SAVE R4
4226 031710 010504 MOV R5,R4 ; R4 POINTS TO DESTINATION ADDRESS
4227 031712 012703 000256' MOV @DEST,R3 ; R3 POINTS TO DEST:
4228 031716 012423 MOV (R4)+,(R3)+ ; LOAD DEST:
4229 031720 012423 MOV (R4)+,(R3)+
4230 031722 012423 MOV (R4)+,(R3)+
4231 031724 012604 MOV (SP)+,R4 ; RESTORE R4
4232 031726 012603 MOV (SP)+,R3 ; RESTORE R3
4233 031730 000207 RTS PC ; AND RETURN
    
```

4235  
 4236  
 4237  
 4238  
 4239  
 4240  
 4241  
 4242  
 4243  
 4244  
 4245  
 4246  
 4247  
 4248  
 4249  
 4250  
 4251  
 4252  
 4253  
 4254  
 4255 031732  
 4256 031732 010046  
 4257 031734 010146  
 4258 031736 010146  
 4259 031740 010546  
 4260  
 4261 031742 012700 000272'  
 4262 031746 012704 000264'  
 4263 031752 012705 000256'  
 4264 031756 012701 000003  
 4265 031762  
 4266 031762 011024  
 4267 031764 012025  
 4268 031766 077103  
 4269  
 4270 031770 012605  
 4271 031772 012604  
 4272 031774 012601  
 4273 031776 012600  
 4274  
 4275 032000 000207  
 4276

```

*****
:
: SUBROUTINE - LDDFLT
:
: THIS SUBROUTINE WILL LOAD THE DEFAULT PHYSICAL ADDRESS
: (TABLE 'DFAULT') INTO BOTH TABLES 'SRC' AND 'DEST'.
:
: INPUTS - NONE
:
: IMPLICIT INPUTS - TABLE 'DFAULT' CONTAINS DEFAULT PHYSICAL ADDR.
:
: OUTPUTS - NONE
:
: PARAMETERS MODIFIED - TABLES 'SRC' AND 'DEST' WILL BE MODIFIED
:
: CALLING SEQUENCE - JSR PC,LDDFLT ;GET DEFAULT ADDRESS DATA
*****
    
```

```

LDDFLT:
    MOV     R0,-(SP)           ; SAVE R0
    MOV     R1,-(SP)           ; SAVE R1
    MOV     R4,-(SP)           ; SAVE R4
    MOV     R5,-(SP)           ; SAVE R5

    MOV     @DFAULT,R0        ; BASE ADDRESS OF DEFAULT ADDRESS
    MOV     @SRC,R4            ; BASE ADDRESS OF SOURCE ADDRESS
    MOV     @DEST,R5          ; BASE ADDRESS OF DEST. ADDRESS
    MOV     @3,R1              ; INIT COUNTER

1$:
    MOV     (R0),(R4)+        ; LOAD ADDRESS
    MOV     (R0)+,(R5)+      ; IN EACH TABLE
    SOB     R1,1$            ; UNTIL DONE

    MOV     (SP)+,R5          ; RESTORE R5
    MOV     (SP)+,R4          ; RESTORE R4
    MOV     (SP)+,R1          ; RESTORE R1
    MOV     (SP)+,R0          ; RESTORE R0

    RTS     PC
    
```

4278  
 4279  
 4280  
 4281  
 4282  
 4283  
 4284  
 4285  
 4286  
 4287  
 4288  
 4289  
 4290  
 4291  
 4292  
 4293  
 4294 032002  
 4295 032002 010346  
 4296 032004 010446  
 4297 032006 012703 000300  
 4298 032012 010504  
 4299 032014 012423  
 4300 032016 012423  
 4301 032020 012423  
 4302 032022 012423  
 4303 032024 012604  
 4304 032026 012603  
 4305 032030 000207

```

*****
:
: SUBROUTINE - LDPCBB
:
: THIS SUBROUTINE MOVES A SELECTED DEFAULT
: PORT CONTROL FUNCTION INTO PCBB.
:
: INPUTS: R5 = ADDRESS OF DEFAULT PORT CONTROL FUNCTION
:
: OUTPUTS: PCBB = SELECTED DEFAULT PORT FUNCTION
:
: CALLING SEQUENCE:
: JSR PC,LDPCBB
*****
    
```

```

LDPCBB:
MOV R3,-(SP) ; SAVE R3
MOV R4,-(SP) ; SAVE R4
MOV #PCBB,R3 ; ADDRESS OF PCBB -> R3
MOV R5,R4 ; R4 = ADDRESS OF DEFAULT FUNCTION
MOV (R4)+,(R3)+ ; LOAD PCBB+0
MOV (R4)+,(R3)+ ; LOAD PCBB+2
MOV (R4)+,(R3)+ ; LOAD PCBB+4
MOV (R4)+,(R3)+ ; LOAD PCBB+6
MOV (SP)+,R4 ; RESTORE R4
MOV (SP)+,R3 ; RESTORE R3
RTS PC ; AND RETURN
    
```

4307  
4308  
4309  
4310  
4311  
4312  
4313  
4314  
4315  
4316  
4317  
4318  
4319  
4320  
4321  
4322  
4323  
4324 032032  
4325 032032 012777 000300' 146170  
4326 032040 012777 000000 146164  
4327 032046 000207

```
*****  
:                                     :  
: SUBROUTINE - LDPCSR                 :  
:                                     :  
: THIS ROUTINE MOVES THE ADDRESS OF PCBB :  
: INTO PCSR2 AND PCSR3.               :  
:                                     :  
: INPUTS:          NONE                :  
:                                     :  
: OUTPUTS:         PCSR2 AND PCSR3 = ADDRESS OF PCBB :  
:                                     :  
: CALLING SEQUENCE:                   :  
: JSR      PC,LDPCSR                   :  
:                                     :  
*****
```

```
LDPCSR:  MOV    @PCBB,@PCSR2           ; ADDRESS OF PCBB -> PCSR2  
         MOV    @ZERO,@PCSR3         ; CLEAR PCSR3  
         RTS    PC                    ; AND RETURN
```

```
4329 ;*****
4330 ;
4331 ; SUBROUTINE - LDPHYA
4332 ;
4333 ; THIS SUBROUTINE WILL MODIFY THE DEFAULT PHYSICAL ADDRESS
4334 ; TABLE AS DETERMINED BY THE DATA IN THE TABLE WHOSE BASE
4335 ; ADDRESS IS PASSED TO THIS ROUTINE.
4336 ;
4337 ; INPUT - R5 - CONTAINS BASE ADDRESS OF TABLE OF NEW ADDRESSES
4338 ;
4339 ; OUTPUT - NONE
4340 ;
4341 ; PARAMETERS MODIFIED - TABLE 'WTPHYA' MAY BE MODIFIED
4342 ;
4343 ; SUBROUTINE CALL - MOV #____,R5 ;GET BASE ADDR. OF NEW ADDR.TABLE
4344 ; JSR PC,LDPHYA ;MODIFY TABLE 'WTPHYA'
4345 ;
4346 ;*****
4347
4348 032050 LDPHYA:
4349 032050 010046 MOV R0,-(SP) ;SAVE R0
4350
4351 032052 012700 012502' MOV #WTPHYA+2,R0 ;POINT TO 2ND ENTRY IN TABLE
4352 032056 012520 MOV (R5)+,(R0)+ ;LOAD
4353 032060 012520 MOV (R5)+,(R0)+ ; NEW
4354 032062 012520 MOV (R5)+,(R0)+ ; ADDRESS
4355
4356 032064 012600 MOV (SP)+,R0 ;RESTORE R0
4357 032066 000207 RTS PC
4358
```

```
4360 ;*****
4361 ;
4362 ; SUBROUTINE - LDRDRB
4363 ;
4364 ; THIS SUBROUTINE MOVES A SELECTED DEFAULT
4365 ; RECEIVE DESCRIPTOR RING INTO RDRB.
4366 ;
4367 ; INPUTS: R5 = ADDRESS OF DATA TO BE MOVED INTO RDRB
4368 ;
4369 ; OUTPUTS: RDRB = SELECTED DEFAULT RECEIVE DESCRIPTOR RING
4370 ;
4371 ; CALLING SEQUENCE:
4372 ; JSR PC,LDRDRB
4373 ;
4374 ;*****
4375 ;
4376 032070 LDRDRB:
4377 032070 010046 MOV R0,-(SP) ; SAVE R0
4378 032072 010346 MOV R3,-(SP) ; SAVE R3
4379 032074 010446 MOV R4,-(SP) ; SAVE R4
4380 032076 012700 000020 MOV #16.,R0 ; LOAD 16 WORDS
4381 032102 012703 000660' MOV #RDRB,R3 ; ADDRESS OF RDRB -> R3
4382 032106 010504 MOV R5,R4 ; R4 = ADDRESS OF DEFAULT RDRB
4383 032110 012423 10$: MOV (R4)+,(R3)+ ; LOAD WORD INTO RDRB
4384 032112 005300 DEC R0 ; DONE ?
4385 032114 001375 BNE 10$ ; NO, KEEP ON LOADING RDRB
4386 032116 012604 MOV (SP)+,R4 ; YES, RESTORE R4
4387 032120 012603 MOV (SP)+,R3 ; RESTORE R3
4388 032122 012600 MOV (SP)+,R0 ; RESTORE R0
4389 032124 000207 RTS PC ; AND RETURN
4390
```

```

4392 :*****
4393 :
4394 :       SUBROUTINE - LDRDRX
4395 :
4396 :       THIS SUBROUTINE MOVES A SELECTED DEFAULT RECEIVE
4397 :       DESCRIPTOR RING INTO RDRBX.
4398 :
4399 :       INPUTS: R5 = ADDRESS OF DATA TO BE MOVED INTO RDRBX
4400 :
4401 :       OUTPUTS:      DRRBX = SELECTED DEFAULT RECEIVE DESCR. RING
4402 :
4403 :       CALLING SEQUENCE:
4404 :                       JSR      PC,LDRDRX
4405 :
4406 :*****
    
```

```

4407
4408 032126
4409 032126 010046
4410 032130 010346
4411 032132 010446
4412 032134 012700 0C0304
4413 032140 012703 001624
4414 032144 010504
4415 032146
4416 032146 012423
4417 032150 005300
4418 032152 001375
4419 032154 012604
4420 032156 012603
4421 032160 012600
4422 032162 000207
4423
    
```

```

LDRDRX:
MOV     R0,-(SP)      ; SAVE R0
MOV     R3,-(SP)      ; SAVE R3
MOV     R4,-(SP)      ; SAVE R4
MOV     #196.,R0      ; LOAD 196 WORDS (49 ENTRIES)
MOV     @RDRX,R3      ; BASE ADDRESS OF RDRBX --> R3
MOV     R5,R4         ; BASE ADDRESS OF DATA --> R4

10$:
MOV     (R4)+,(R3)+   ; LOAD WORD INTO RDRBX
DEC     R0             ; DONE?
BNE     10$           ; NO, KEEP ON LOADING
MOV     (SP)+,R4      ; YES, RESTORE R4
MOV     (SP)+,R3      ; RESTORE R3
MOV     (SP)+,R0      ; RESTORE R0
RTS     PC            ; AND RETURN
    
```

```

4425 ;*****
4426 ;
4427 ; SUBROUTINE - LDTDRB
4428 ;
4429 ; THIS SUBROUTINE MOVES A SELECTED DEFAULT
4430 ; TRANSMIT DESCRIPTOR RING INTO TDRB.
4431 ;
4432 ; INPUTS: R5 = ADDRESS OF DATA TO BE MOVED INTO TDRB
4433 ;
4434 ; OUTPUTS: TDRB = SELECTED DEFAULT TRANSMIT DESCRIPTOR RING
4435 ;
4436 ; CALLING SEQUENCE:
4437 ; JSR PC,LDTDRB
4438 ;
4439 ;*****
    
```

```

4440
4441 032164 LDTDRB:
4442 032164 010046 MOV R0,-(SP) ; SAVE R0
4443 032166 010346 MOV R3,-(SP) ; SAVE R3
4444 032170 010446 MOV R4,-(SP) ; SAVE R4
4445 032172 012700 000020 MOV #16,R0 ; LOAD 16 WORDS
4446 032176 012703 000620' MOV #TDRB,R3 ; ADDRESS OF TDRB -> R3
4447 032202 010504 MOV R5,R4 ; R4 = ADDRESS OF DEFAULT TDRB
4448 032204 012423 10$: MOV (R4)+,(R3)+ ; LOAD WORD INTO TDRB
4449 032206 005300 DEC R0 ; DONE ?
4450 032210 001375 BNE 10$ ; NO, KEEP ON LOADING TDRB
4451 032212 012604 MOV (SP)+,R4 ; YES, RESTORE R4
4452 032214 012603 MOV (SP)+,R3 ; RESTORE R3
4453 032216 012600 MOV (SP)+,R0 ; RESTORE R0
4454 032220 000207 RTS PC ; AND RETURN
    
```



4456  
 4457  
 4458  
 4459  
 4460  
 4461  
 4462  
 4463  
 4464  
 4465  
 4466  
 4467  
 4468  
 4469  
 4470  
 4471  
 4472

```

;*****
;
;   SUBROUTINE - LDTDRX
;
;   THIS SUBROUTINE MOVES A SELECTED DEFAULT
;   TRANSMIT DESCRIPTOR RING INTO TDRX.
;
;   INPUTS:          R5 = ADDRESS OF DATA TO BE MOVED INTO TDRX
;
;   OUTPUTS:         TDRX = SELECTED DEFAULT TRANSMIT DESCRIPTOR RING
;
;   CALLING SEQUENCE:
;                   JSR     PC,LDTDRX
;*****
    
```

4473 032222 010046  
 4474 032222 010346  
 4475 032224 010346  
 4476 032226 010446  
 4477 032230 012700 000304  
 4478 032234 012703 001014'  
 4479 032240 010504  
 4480 032242 012423  
 4481 032244 005300  
 4482 032246 001375  
 4483 032250 012604  
 4484 032252 012603  
 4485 032254 012600  
 4486 032256 000207

```

LDTDRX:
MOV     R0,-(SP)           ; SAVE R0
MOV     R3,-(SP)           ; SAVE R3
MOV     R4,-(SP)           ; SAVE R4
MOV     #196.,R0           ; LOAD 196 WORDS (49 ENTRIES)
MOV     #TDRX,R3           ; ADDRESS OF TDRX -> R3
MOV     R5,R4              ; R4 = ADDRESS OF DEFAULT TDRB
10$:   MOV     (R4)+,(R3)+   ; LOAD WORD INTO TDRB
DEC     R0                  ; DONE ?
BNE     10$                 ; NO, KEEP ON LOADING TDRB
MOV     (SP)+,R4           ; YES, RESTORE R4
MOV     (SP)+,R3           ; RESTORE R3
MOV     (SP)+,R0           ; RESTORE R0
RTS     PC                  ; AND RETURN
    
```

4488  
 4489  
 4490  
 4491  
 4492  
 4493  
 4494  
 4495  
 4496  
 4497  
 4498  
 4499  
 4500  
 4501  
 4502  
 4503  
 4504  
 4505  
 4506 032260  
 4507 032260 010146  
 4508 032262 010346  
 4509 032264 010446  
 4510 032266 010001  
 4511 032270 012703 000310'  
 4512 032274 010504  
 4513 032276 012423  
 4514 032300 005301  
 4515 032302 001375  
 4516 032304 012604  
 4517 032306 012603  
 4518 032310 012601  
 4519 032312 000207

```

;*****
;
;      SUBROUTINE - LDUDBB
;
;      THIS ROUTINE MOVES A SELECTED DEFAULT
;      DATA STRUCTURE INTO UDBB.
;
;      INPUTS:          R5 = ADDRESS OF DATA TO BE MOVED INTO UDBB
;                      R0 = NUMBER OF WORDS TO BE MOVED
;
;      OUTPUTS:         UDBB = SELECTED DEFAULT DATA STRUCTURE
;
;      CALLING SEQUENCE:
;      JSR      PC,LDUDBB
;*****
    
```

```

LDUDBB:
        MOV     R1,-(SP)           ; SAVE R1
        MOV     R3,-(SP)           ; SAVE R3
        MOV     R4,-(SP)           ; SAVE R4
        MOV     R0,R1             ; R1= NUMBER OF WORDS TO BE MOVED
        MOV     @UDBB,R3          ; ADDRESS OF UDBB -> R3
        MOV     R5,R4             ; R4= ADDRESS OF DATA TO BE MOVED
10$:    MOV     (R4)+,(R3)+        ; LOAD WORD INTO UDBB
        DEC     R1                 ; DONE ?
        BNE    10$                ; NO, KEEP ON LOADING
        MOV     (SP)+,R4          ; YES, RESTORE R4
        MOV     (SP)+,R3          ; RESTORE R3
        MOV     (SP)+,R1          ; RESTORE R1
        RTS     PC                 ; AND RETURN
    
```

```
4521  
4522 ;*****  
4523 ;  
4524 ; NOTE: MAY BE ABLE TO DELETE THIS FROM FINAL PRODUCT  
4525 ;  
4526 ; SUBROUTINE - LDXCRC  
4527 ;  
4528 ; THIS SUBROUTINE LOADS XCRC WITH EXPECTED CRC DATA.  
4529 ;  
4530 ; INPUTS: R5 = ADDRESS OF EXPECTED DATA  
4531 ;  
4532 ; OUTPUTS: XCRC TABLE = EXPECTED CRC DATA  
4533 ;  
4534 ; CALLING SEQUENCE:  
4535 ; JSR PC,LDXCRC  
4536 ;  
4537 ;*****
```

```
4538  
4539 032314 LDXCRC:  
4540 032314 010346 MOV R3,-(SP) ; SAVE R3  
4541 032316 010446 MOV R4,-(SP) ; SAVE R4  
4542 032320 012704 016574' MOV @XCRC,R4 ; R4 POINTS TO XCRC  
4543 032324 010503 MOV R5,R3 ; R3 POINTS TO DATA  
4544 032326 012324 MOV (R3)+,(R4)+ ; LOAD XCRC TABLE  
4545 032330 012324 MOV (R3)+,(R4)+  
4546 032332 012604 MOV (SP)+,R4 ; RESTORE R4  
4547 032334 012603 MOV (SP)+,R3 ; RESTORE R3  
4548 032336 000207 RTS PC ; AND RETURN
```

4550  
 4551  
 4552  
 4553  
 4554  
 4555  
 4556  
 4557  
 4558  
 4559  
 4560  
 4561  
 4562  
 4563  
 4564  
 4565

```

*****
:
: SUBROUTINE - LDXRDR
:
: THIS SUBROUTINE LOADS XRDRBO WITH EXPECTED RDRB DATA.
:
: INPUTS:          R5 = ADDRESS OF EXPECTED DATA
:
: OUTPUTS:         XRDRBO TABLE = EXPECTED RDRB DATA
:
: CALLING SEQUENCE:
:                   JSR      PC,LDXRDR
*****
    
```

4566 032340  
 4567 032340 010346  
 4568 032342 010446  
 4569 032344 012704 016530'  
 4570 032350 010503  
 4571 032352 012324  
 4572 032354 012324  
 4573 032356 012324  
 4574 032360 012324  
 4575 032362 012604  
 4576 032364 012603  
 4577 032366 000207

```

LDXRDR:
MOV     R3,-(SP)           ; SAVE R3
MOV     R4,-(SP)           ; SAVE R4
MOV     #XRDRBO,R4        ; R4 POINTS TO XRDRBO
MOV     R5,R3              ; R3 POINTS TO DATA
MOV     (R3)+,(R4)+       ; LOAD XRDRBO TABLE
MOV     (R3)+,(R4)+
MOV     (R3)+,(R4)+
MOV     (R3)+,(R4)+
MOV     (SP)+,R4          ; RESTORE R4
MOV     (SP)+,R3          ; RESTORE R3
RTS     PC                 ; AND RETURN
    
```

4579  
 4580  
 4581  
 4582  
 4583  
 4584  
 4585  
 4586  
 4587  
 4588  
 4589  
 4590  
 4591  
 4592  
 4593  
 4594  
 4595 032370  
 4596 032370 010346  
 4597 032372 010446  
 4598 032374 012700 000004  
 4599 032400 012704 016550  
 4600 032404 010503  
 4601 032406  
 4602 032406 012324  
 4603 032410 005300  
 4604 032412 001375  
 4605  
 4606 032414 012604  
 4607 032416 012603  
 4608 032420 000207

```

*****
:
:      SUBROUTINE - LDXTDR
:
:      THIS SUBROUTINE LOADS XTDRBO WITH EXPECTED TORB DATA.
:
:      INPUTS:          R5 = ADDRESS OF EXPECTED DATA
:
:      OUTPUTS:         XTDRBO TABLE = EXPECTED TORB DATA
:
:      CALLING SEQUENCE:
:          JSR      PC,LDXTDR
*****
    
```

```

LDXTDR:
      MOV      R3,-(SP)          ; SAVE R3
      MOV      R4,-(SP)          ; SAVE R4
      MOV      #4,R0             ; LOOP COUNT
      MOV      @XTDRBO,R4        ; R4 POINTS TO XTDRBO
      MOV      R5,R3             ; R3 POINTS TO DATA

10$:
      MOV      (R3)+,(R4)+       ; LOAD XTDRBO TABLE
      DEC      R0                 ; REDUCE LOOP COUNT
      BNE     10$                 ; LOOP AGAIN IF NOT COMPLETED

      MOV      (SP)+,R4          ; RESTORE R4
      MOV      (SP)+,R3          ; RESTORE R3
      RTS      PC                 ; AND RETURN
    
```

4610  
 4611  
 4612  
 4613  
 4614  
 4615  
 4616  
 4617  
 4618  
 4619  
 4620  
 4621  
 4622  
 4623  
 4624  
 4625  
 4626  
 4627  
 4628  
 4629  
 4630  
 4631  
 4632 032422  
 4633 032422 010446  
 4634 032424 017704 145574  
 4635 032430 032704 020000  
 4636 032434 001407  
 4637 032436 010437 016514'  
 4638 032442 017737 145560 016516'  
 4639 032450 000261  
 4640 032452 000401  
 4641 032454 000241  
 4642 032456 012604  
 4643 032460 000207

```

*****
:
: SUBROUTINE NORXI
:
: THIS SUBROUTINE VERIFIES THE RXI BIT IS NOT SET.
:
: INPUTS: NONE
:
: OUTPUTS: IF RXI NOT SET ( RXI = 0 )
:          THEN CARRY = 0
:
:          IF RXI IS SET ( RXI = 1 )
:          THEN CARRY = 1
:          PCSRO -> EPCSRO
:          PCSR1 -> EPCSR1
:
: CALLING SEQUENCE:
: JSR PC,NORXI
*****
    
```

```

NORXI:
MOV R4,-(SP) ; SAVE R4
MOV @PCSRO,R4 ; PCSRO -> R4
BIT @RXI,R4 ; RXI = 0 ?
BEQ 10$ ; YES
MOV R4,EPCSRO ; NO, PCSRO -> EPCSRO
MOV @PCSR1,EPCSR1 ; PCSR1 -> EPCSR1
SEC ; SET CARRY
BR 20$
10$: CLC ; CLEAR CARRY
20$: MOV (SP)+,R4 ; RESTORE R4
RTS PC ; AND RETURN
    
```

```

4645 ;*****
4646 ;
4647 ; SUBROUTINE - ROMCRC (Used specifically in Read Int.ROM Test)
4648 ;
4649 ; This subroutine calculates a 16 bit CRC on a block of data.
4650 ; Used explicitly for ROM CRC calculation.
4651 ;
4652 ; IMPLICIT
4653 ; INPUTS: RBUF = BASE ADDRESS OF DATA BLOCK
4654 ; BYTCNT = DATA BLOCK BYTE COUNT
4655 ;
4656 ; OUTPUTS: R4,R3 = CRC
4657 ; XCRC = BASE ADDRESS OF CRC STORAGE TABLE,
4658 ; CONTAINING UPDATED DATA FROM R4,R3
4659 ;
4660 ; CALLING SEQUENCE:
4661 ; JSR PC,ROMCRC ;GO CALCULATE CRC
4662 ;
4663 ;*****
    
```

```

4664
4665 032462 ROMCRC:
4666 032462 010046 MOV R0,-(SP) ; SAVE R0
4667 032464 010146 MOV R1,-(SP) ; SAVE R1
4668 032466 010246 MOV R2,-(SP) ; SAVE R2
4669 032470 010546 MOV R5,-(SP) ; SAVE R5
4670
4671 032472 012701 006440' MOV #RBUF,R1 ; GET BASE ADDRESS OF DATA BLOCK
4672 032476 013702 016560' MOV BYTCNT,R2 ; GET DATA BLOCK BYTE COUNT
4673 032502 012700 016574' MOV #XCRC,R0 ; GET BASE ADDRESS OF INITIAL CRC
4674 032506 012003 MOV (R0)+,R3 ; LOAD INITIAL
4675 032510 011004 MOV (R0),R4 ; CRC
4676 032512
4677 032512 112105 1$: MOVB (R1)+,R5 ; GET NEXT CHARACTER
4678 032514 004737 031160' JSR PC,GETCRC ; CALCULATE CRC
4679 032520 077204 SOB R2,1$ ; DO NEXT CHARACTER IF NOT FINISHED
4680
4681 032522 012700 016574' MOV #XCRC,R0 ; POINT TO BASE STORAGE ADDRESS
4682 032526 010320 MOV R3,(R0)+ ; UPDATE 1ST WORD
4683 032530 010410 MOV R4,(R0) ; UPDATE 2ND WORD
4684
4685 032532 012605 MOV (SP)+,R5 ; RESTORE R5
4686 032534 012602 MOV (SP)+,R2 ; RESTORE R2
4687 032536 012601 MOV (SP)+,R1 ; RESTORE R1
4688 032540 012600 MOV (SP)+,R0 ; RESTORE R0
4689
4690 032542 000207 RTS PC
4691
4692
    
```

4694  
 4695  
 4696  
 4697  
 4698  
 4699  
 4700  
 4701  
 4702  
 4703  
 4704  
 4705  
 4706  
 4707  
 4708  
 4709  
 4710  
 4711  
 4712  
 4713  
 4714  
 4715  
 4716  
 4717  
 4718  
 4719  
 4720  
 4721  
 4722  
 4723  
 4724  
 4725  
 4726  
 4727  
 4728  
 4729  
 4730  
 4731 032544  
 4732 032544 010046  
 4733 032546 010146  
 4734 032550 010346  
 4735 032552 010446  
 4736 032554 010546  
 4737  
 4738  
 4739  
 4740 032556 012703 006440'  
 4741 032562 012700 000036  
 4742 032566 005023  
 4743 032570 005300  
 4744 032572 001375  
 4745  
 4746  
 4747  
 4748  
 4749 032574 012705 000264'  
 4750 032600 012703 002436'

```

*****
SUBROUTINE - SETBF (Used specifically in Int. Loopback Length Err Tst)

THIS ROUTINE SETS UP THE TRANSMIT AND RECEIVE BUFFERS
FOR A DATA BYTE LENGTH DETERMINED BY VALUE PASSED IN
R2. CRC WILL ALSO BE CALCULATED, SAVED, AND IF REQUIRED,
APPENDED TO END OF DATA IF PARAMETER 'DOCRC' SO INDICATES.

A. CLEAR RECEIVE BUFFER RBUF
B. LOAD TRANSMIT BUFFER TBUF (DEST. ADDRESS, SOURCE ADDRESS, DATA)
C. CALCULATE CRC IF CRC FLAG SET

INPUTS:          DOCRC = 0 THEN NO CRC CALCULATION
                  1 THEN CALCULATE CRC, SAVE, AND APPEND
                  -1 THEN CALCULATE CRC, SAVE, DO NOT APPEND

                  R2 = number of data bytes

IMPLICIT INPUTS:
                  DEST: = DESTINATION ADDRESS

OUTPUTS:         RBUF IS CLEARED
                  TBUF IS SET UP FOR TRANSMIT
                  R3,R4 AND ALSO 'XCRC' CONTAIN
                  CRC IF 'DOCRC' WAS SET.

PARAMETERS MODIFIED: 'DOCRC' WILL ALWAYS BE CLEARED ON EXIT
                    R3,R4 MODIFIED IF 'DOCRC' SET.

CALLING SEQUENCE:
                    INSURE 'DOCRC' FLAG CONTAINS CORRECT DATA

                    JSR    PC,SETBF
*****

SETBF:
    MOV     R0,-(SP)          ;SAVE R0
    MOV     R1,-(SP)          ;SAVE R1
    MOV     R3,-(SP)          ;SAVE R3
    MOV     R4,-(SP)          ;SAVE R4
    MOV     R5,-(SP)          ;SAVE R5

;CLEAR 'RBUF' 30. WORDS

    MOV     #RBUF,R3          ;POINT TO RBUF
    MOV     #30.,R0           ;COUNT = 60 BYTES
10$:    CLR     (R3)+          ;CLEAR BUFFER
        DEC     R0            ;DONE?
        BNE    10$           ;NO

;SET UP TRANSMIT BUFFER 'TBUF'
;LOAD DESTINATION ADDRESS

    MOV     #SRC,R5           ;POINT TO DESTINATION ADDRESS
    MOV     #TBUF,R3          ;POINT TO TBUF
    
```



```

4751 032604 012523          MOV    (R5)+,(R3)+      ;LOAD DESTINATION ADDRESS
4752 032606 012523          MOV    (R5)+,(R3)+      ;
4753 032610 012523          MOV    (R5)+,(R3)+      ;
4754
4755          ;LOAD SOURCE ADDRESS
4756
4757 032612 012705 000264'   MOV    #SRC,R5          ;LOAD FOR LATER COMPARISON
4758 032616 012523          MOV    (R5)+,(R3)+      .
4759 032620 012523          MOV    (R5)+,(R3)+      ;
4760 032622 012523          MOV    (R5)+,(R3)+      ;
4761
4762          ;SET TYPE FIELD
4763
4764 032624 012723 002540     MOV    #2540,(R3)+      ;TYPE FIELD = DIAGNOSTICS
4765
4766          ;LOAD DATA FIELD
4767
4768 032630 010200          MOV    R2,R0           ;NUMBER OF WORDS
4769 032632 012705 016626'   MOV    #PATRN1,R5      ;POINT TO DATA PATTERN
4770 032636 012523          20$: MOV    (R5)+,(R3)+  ;LOAD DATA PATTERN
4771 032640 005300          DEC    R0              ;DONE?
4772 032642 001375          BNE   20$             ;NO
4773
4774          ;CALCULATE CRC AND SAVE IN 'XCRC'
4775
4776 032644 010146          MOV    R1,-(SP)        ;SAVE R1
4777 032646 010246          MOV    R2,-(SP)        ;SAVE R2
4778 032650 010346          MOV    R3,-(SP)        ;SAVE R3
4779 032652 006302          ASL   R2              ;MULTIPLY BY 2 FOR BYTES
4780 032654 062702 000020     ADD   #16,R2          ;ADD HEADER
4781 032660 012701 002436'   MOV    #TBUF,R1        ;BASE ADDR OF TRANSMIT BUFFER
4782 032664 012703 177777     MOV    #-1,R3          ;INIT CRC
4783 032670 012704 177777     MOV    #-1,R4          ;INIT CRC
4784 032674 004737 026506'   JSR   PC,BLKCRC        ;CALCULATE AND SAVE CRC
4785 032700 012603          MOV    (SP)+,R3        ;RESTORE R3
4786 032702 012602          MOV    (SP)+,R2        ;RESTORE R2
4787 032704 012601          MOV    (SP)+,R1        ;RESTORE R1
4788
4789          ;IF CRC FLAG SET, APPEND CRC
4790
4791 032706 005737 016562'   TST   D0CRC           ;GENERATE CRC?
4792 032712 001402          BEQ   30$             ;NO, SKIP CRC GENERATION
4793 032714 010423          MOV    R4,(R3)+        ;APPEND CRC LOW WORD
4794 032716 010313          MOV    R3,(R3)         ;APPEND CRC HIGH WORD
4795 30$:
4796 032720 012605          MOV    (SP)+,R5        ;RESTORE R5
4797 032722 012604          MOV    (SP)+,R4        ;RESTORE R4
4798 032724 012603          MOV    (SP)+,R3        ;RESTORE R3
4799 032726 012601          MOV    (SP)+,R1        ;RESTORE R1
4800 032730 012600          MOV    (SP)+,R0        ;RESTORE R0
4801 032732 000207          RTS   PC              ;RETURN TO CALLING ROUTINE
4802
    
```

4804  
 4805  
 4806  
 4807  
 4808  
 4809  
 4810  
 4811  
 4812  
 4813  
 4814  
 4815  
 4816  
 4817  
 4818  
 4819  
 4820  
 4821  
 4822  
 4823  
 4824  
 4825  
 4826  
 4827  
 4828  
 4829  
 4830  
 4831  
 4832

```

:*****
:
:   SUBROUTINE PNTID
:
:   PRINTS THE NAME OF EACH TEST THAT IS RUN, IF PRINT FLAG
:   SET, AND IS 1ST LOOP THROUGH TEST.
:
:   INPUTS: R4 = POINTER TO TEST NAME MESSAGE
:
:   OUTPUT: IF PRINT FLAG SET, TEST NAME WILL BE PRINTED
:
:   CALL:  MOV   #MSGNO,R4      ;GET ADDRESS OF MESSAGE
:         JSR   PC,PNTID      ;PRINT TEST NAME
:*****
PNTID:
    TST   PRNTIT              ;PRINT THE TEST NAME?
    BEQ   10$                ;NO
    MOV   R4,R0              ;SETUP FOR PRINT
    PRINTF #TSTFMT,R0       ;PRINT TEST NAME
                                MOV   R0,-(SP)
                                MOV   #TSTFMT,-(SP)
                                MOV   #2,-(SP)
                                MOV   SP,R0
                                TRAP  C$PNTF
                                ADD   #6,SP
10$:
    RTS   PC                  ;RETURN TO CALLING PROGRAM

TSTFMT:.ASCIZ 'S#T#A TEST '

.EVEN
    
```

032734 005737 016620'  
 032740 001412  
 032742 010400  
 032744 010046  
 032746 012746 032770'  
 032752 012746 000002  
 032756 010600  
 032760 104417  
 032762 062706 000006  
 032766 000207  
 032770 045 123 045  
 032773 124 045 101  
 032776 040 124 105  
 033001 123 124 040  
 033004 040 000

```

4834 :*****
4835 :
4836 :   SUBROUTINE - SETBUF
4837 :
4838 :   THIS ROUTINE SETS UP THE TRANSMIT AND RECEIVE BUFFERS. THE
4839 :   NUMBER OF BYTES IN A PACKET IS DETERMINED BY THE CONTENTS
4840 :   PASSED IN 'BYTCNT'. IN ANY CASE, A PACKET LENGHT WILL BE NO
4841 :   MORE THAN 32 BYTES, OR 36 INCLUDING CRC IF 'DOCRC' SET.
4842 :
4843 :   A. LOAD TRANSMIT BUFFER TBUF (DEST. ADDR,SOURCE ADDR,TYPE,DATA)
4844 :   B. APPEND CRC IF 'DOCRC' FLAG SET
4845 :
4846 :   INPUTS:           DOCRC = 0 THEN NO CRC
4847 :                     1 THEN CALCULATE CRC AND APPEND
4848 :   BYTCNT= # OF DATA BYTES IN PACKET
4849 :
4850 :   IMPLICIT INPUTS:
4851 :                     SRC: = SOURCE ADDRESS
4852 :                     DEST: = DESTINATION ADDRESS
4853 :
4854 :   OUTPUTS:
4855 :                     TBUF IS SET UP FOR TRANSMIT
4856 :
4857 :   PARAMETERS MODIFIED: 'DOCRC' WILL ALWAYS BE CLEARED ON EXIT
4858 :
4859 :   CALLING SEQUENCE:
4860 :                     INSURE 'DOCRC' FLAG CONTAINS CORRECT DATA
4861 :
4862 :                     MOV     #Z,BYTCNT      ;NUMBER OF BYTES THIS PACKET
4863 :                     JSR     PC.SETBUF
4864 :
4865 :*****
4866 :
4867 :
4868 033006 SETBUF:
4869 033006 010046      MOV     R0,-(SP)      ; SAVE R0
4870 033010 010146      MOV     R1,-(SP)      ; SAVE R1
4871 033012 010246      MOV     R2,-(SP)      ; SAVE R2
4872 033014 010346      MOV     R3,-(SP)      ; SAVE R3
4873 033016 010446      MOV     R4,-(SP)      ; SAVE R4
4874 033020 010546      MOV     R5,-(SP)      ; SAVE R5
4875
4876      ;SET UP TRANSMIT BUFFER TBUF
4877      ;LOAD DESTINATION ADDRESS
4878
4879 033022 012705 000256'  MOV     #DEST,R5      ; POINT TO DESTINATION ADDRESS
4880 033026 012701 002436'  MOV     #TBUF,R1      ; POINT TO TBUF
4881 033032 012521          MOV     (R5)+,(R1)+   ; LOAD DESTINATION ADDRESS
4882 033034 012521          MOV     (R5)+,(R1)+
4883 033036 012521          MOV     (R5)+,(R1)+
4884
4885      ;LOAD SOURCE ADDRESS
4886
4887 033040 012705 000264'  MOV     #SRC,R5       ; LOAD FOR LATER COMPARISON
4888 033044 012521          MOV     (R5)+,(R1)+
4889 033046 012521          MOV     (R5)+,(R1)+
4890 033050 012521          MOV     (R5)+,(R1)+
    
```

```

4891
4892           ;SET TYPE FIELD
4893
4894 033052 012721 002540           MOV     #2540,(R1)+           ; ENTER DIAGNOSTIC IC IN TYPE FIELD
4895
4896           ;LOAD DATA FIELD (LENGTH DEPENDENT ON CONTENTS OF 'BYTCNT')
4897
4898 033056 013700 016560'          MOV     BYTCNT,R0           ; BYTE COUNT
4899 033062 012705 016626'          MOV     #PATRN1,R5         ; POINT TO DATA PATTERN
4900 033066 012521                    20$:   MOV     (R5)+,(R1)+       ; LOAD DATA PATTERN
4901 033070 005300                    DEC     R0                 ; DONE ?
4902 033072 002375                    BGE     20$               ; NO
4903
4904           ;CALCULATE CRC AND SAVE IN 'XCRC'
4905
4906 033074 010146                    MOV     R1,-(SP)          ; SAVE R1
4907 033076 010246                    MOV     R2,-(SP)          ; SAVE R2
4908 033100 013702 016560'          MOV     BYTCNT,R2         ; GET DATA BYTE COUNT
4909 033104 006302                    ASL     R2                 ; ALIGN
4910 033106 005737 016562'          TST     DOCRC             ; ADD CRC TO PACKET?
4911 033112 001003                    BNE     22$               ; YES
4912 033114 062702 000016          ADD     #14.,R2           ; NO
4913 033120 000402                    BR      25$               ; SKIP NO ADD CRC DATA
4914 033122 062702 000020          22$:   ADD     #16.,R2         ; WILL ADD CRC TO PACKET
4915 033126 012701 002436'          25$:   MOV     #TBUF,R1         ; BASE ADDR OF TRANSMIT BUFFER
4916 033132 012703 177777          MOV     #-1,R3            ; INIT CRC
4917 033136 012704 177777          MOV     #-1,R4            ; INIT CRC
4918 033142 004737 026506'          JSR     PC,BLKCRC         ; CALCULATE AND SAVE CRC
4919 033146 012602                    MOV     (SP)+,R2         ; RESTORE R2
4920 033150 012601                    MOV     (SP)+,R1         ; RESTORE R1
4921
4922           ;IF CRC FLAG SET, APPEND CRC, THEN SAVE EXPECTED CRC IN 'XCRC' TABLE
4923
4924 033152 005737 016562'          TST     DOCRC             ;APPEND CRC?
4925 033156 001402                    BEQ     30$               ;NO, SKIP APPENDING CRC
4926 033160 010421                    MOV     R4,(R1)+         ;APPEND CRC LOW WORD
4927 033162 010321                    MOV     R3,(R1)+         ;APPEND CRC HIGH WORD
4928
4929
4930 033164                    30$:   CLR     DOCRC             ;INSURE CRC FLAG IS CLEARED
4931 033164 005037 016562'
4932
4933 033170 012605                    MOV     (SP)+,R5         ; RESTORE R5
4934 033172 012604                    MOV     (SP)+,R4         ; RESTORE R4
4935 033174 012603                    MOV     (SP)+,R3         ; RESTORE R3
4936 033176 012602                    MOV     (SP)+,R2         ; RESTORE R2
4937 033200 012601                    MOV     (SP)+,R1         ; RESTORE R1
4938 033202 012600                    MOV     (SP)+,R0         ; RESTORE R0
4939 033204 000207                    RTS     PC                ; AND RESTORE
    
```

```

4941 :*****
4942 :
4943 :   SUBROUTINE - SRCOST
4944 :
4945 :   THIS SUBROUTINE WILL INDEPENDENTLY, LOAD BOTH TABLES
4946 :   'SRC' AND 'DEST' WITH PHYSICAL ADDRESSES OBTAINED FROM
4947 :   TABLE ADDRESSES PASSED TO THIS ROUTINE.
4948 :
4949 :   INPUT - R1  CONTAINS ADDRESS OF TABLE TO LOAD INTO 'SRC'
4950 :            R2 - CONTAINS ADDRESS OF TABLE TO LOAD INTO 'DEST'
4951 :
4952 :   OUTPUT  NONE
4953 :
4954 :   PARAMETERS MODIFIED - TABLES 'SRC' AND 'DEST' MAY BE CHANGED
4955 :
4956 :   CALLING SEQUENCE - MOV  #____,R1      ; ADDR. OF TABLE TO LOAD 'SRC'
4957 :                     MOV  #____,R2      ; ADDR. OF TABLE TO LOAD 'DEST'
4958 :                     JSR  PC,SRCOST     ; LOAD ADDRESS TABLES
4959 :
4960 :*****
    
```

```

4961 SRCOST:
4962 033206      MOV    R0,-(SP)          ;SAVE R0
4963 033206 010046      MOV    R1,-(SP)          ;SAVE R1
4964 033210 010146      MOV    R2,-(SP)          ;SAVE R2
4965 033212 010246      MOV    R4,-(SP)          ;SAVE R4
4966 033214 010446      MOV    R5,-(SP)          ;SAVE R5
4967 033216 010546
4968
4969 033220 012704 000264'      MOV    #SRC,R4          ;GET BASE ADDR. OF TABLE 'SRC'
4970 033224 012705 000256'      MOV    #DEST,R5        ;GET BASE ADDR. OF TABLE 'DEST'
4971 033230 012700 000003      MOV    #3,R0          ;INIT COUNTER
4972
4973 033234 012124      1$:  MOV    (R1)+,(R4)+        ;LOAD BOTH
4974 033236 012225      MOV    (R2)+,(R5)+        ; ADDRESS TABLES
4975 033240 077003      SOB    R0,1$          ;UNTIL DONE
4976
4977 033242 012605      MOV    (SP)+,R5        ;RESTORE R5
4978 033244 012604      MOV    (SP)+,R4        ;RESTORE R4
4979 033246 012602      MOV    (SP)+,R2        ;RESTORE R2
4980 033250 012601      MOV    (SP)+,R1        ;RESTORE R1
4981 033252 012600      MOV    (SP)+,R0        ;RESTORE R0
4982
4983 033254 000207      RTS    PC
    
```

4985  
 4986  
 4987  
 4988  
 4989  
 4990  
 4991  
 4992  
 4993  
 4994  
 4995  
 4996  
 4997  
 4998  
 4999  
 5000  
 5001  
 5002  
 5003  
 5004  
 5005  
 5006  
 5007  
 5008  
 5009  
 5010  
 5011  
 5012  
 5013  
 5014  
 5015  
 5016  
 5017  
 5018  
 5019  
 5020  
 5021  
 5022  
 5023  
 5024  
 5025  
 5026  
 5027  
 5028  
 5029  
 5030  
 5031  
 5032

```

*****
:
: Subroutine - SRWRAM
:
: THIS SUBROUTINE SETS UP FOR EITHER A LOAD OR A DUMP
: OF A 1K SEGMENT OF INTERNAL RAM. LOAD OR DUMP FUNCTION
: IS DETERMINED BY VALUE PASSED IN R5.
:
: INPUTS: R5 - IF R5 = 20 THEN DUMP MEMORY
:           IF R5 = 21 THEN LOAD MEMORY
:           R1  CONTAINS NUMBER OF 1K RAM BLOCKS TO ACCESS
:           R3  CONTAINS BASE ADDR OF 1K RAM TO R/W
:
: OUTPUTS:      NONE
:
: CALLING SEQUENCE:      MOV FUNCTION,R5      ;GET VALUE OF FUNCTION
:                        JSR   PC,SRWRAM      ;EXECUTE
*****
    
```

```

SRWRAM:
MOV     R2,-(SP)      ;SAVE R2
MOV     R5,R2        ;SAVE R5
JSR     PC,LDPCBB    ;LOAD FUNCTION -> PCBB (USES R5)
MOV     #UDB11A,R5  ;DEFAULT UDBB
MOV     #5,R0        ;FOUR WORDS
JSR     PC,LDUDBB    ;LOAD INTO UDBB
CMP     #21,(R2)     ;IS THIS A LOAD MEMORY?
BEQ     1$          ;YES
MOV     #RBUF,UDBB+2 ;THIS IS A READ MEMORY
BR      2$          ;SKIP SETTING FOR TBUF
MOV     #TBUF,UDBB+2 ;THIS IS A LOAD MEMORY
MOV     (R3),UDBB+6  ;LOAD LINK ADDR -> UDBB+6
TST     EAFLAG      ;NEED TO SET EXT ADDR BITS?
BNE     3$          ;YES
CMP     #176000,(R3) ;TIME TO SET 'EAFLAG'?
BNE     4$          ;NO
INC     EAFLAG      ;YES
BR      4$          ;BUT, ONLY NEXT TIME
MOV     #1,UDBB+10   ;EXT ADDRESS BIT
CMP     #1,R1        ;IS THIS LAST 1K?
BNE     5$          ;NO
MOV     #1200,UDBB   ;YES, ONLY READ 1200(8) WORDS
MOV     (SP)+,R2    ;RESTORE R3
RTS     PC           ;RETURN TO CALLING PROGRAM
    
```

5034  
5035 ;\*\*\*\*\*  
5036 ;  
5037 ; THIS ROUTINE TURNS THE CLOCK OFF  
5038 ;  
5039 ; INPUT: NONE  
5040 ;  
5041 ; OUTPUT: NONE, ON EXIT SUBROUTINE, CLOCK IS OFF  
5042 ;  
5043 ; CALL: JSR PC,TIMOFF  
5044 ;  
5045 ;\*\*\*\*\*

5046  
5047 033402 TIMOFF:  
5048 033402 005077 144640 CLR @CLKCSR ;CLEAR THE INTERRUPT ENABLE  
5049 033406 SETPRI @PRI07 ;UP THE PRIORITY  
033406 012700 000340 MOV @PRI07,R0  
033412 104441 TRAP C:SPRI  
5050 033414 000207 RTS PC ;RETURN TO CALLING ROUTINE

5051 ;\*\*\*\*\*  
5052 ;  
5053 ; THIS SUBROUTINE TURNS ON THE CLOCK  
5054 ;  
5055 ; INPUT: NONE  
5056 ;  
5057 ; OUTPUT: NONE,ON SUBROUTINE EXIT, CLOCK IS ON  
5058 ;  
5059 ; CALL: JSR PC,TIMON  
5060 ;  
5061 ;\*\*\*\*\*

5062  
5063  
5064 033416 TIMON:  
5065 033416 SETPRI @PRI05 ;SET PROCESSOR PRIORITY TO 5  
033416 012700 000240 MOV @PRI05,R0  
033422 104441 TRAP C:SPRI  
5066 033424 012777 000100 144614 MOV @IE,@CLKCSR ;ENABLE CLOCK INTERRUPTS  
5067 033432 000207 RTS PC ;RETURN TO CALLING ROUTINE  
5068  
5069

5071  
 5072  
 5073  
 5074  
 5075  
 5076  
 5077  
 5078  
 5079  
 5080  
 5081  
 5082  
 5083  
 5084  
 5085  
 5086  
 5087  
 5088  
 5089  
 5090  
 5091  
 5092 033434  
 5093 033434 010446  
 5094 033436 004737 030346'  
 5095 033442 112777 000000 144554  
 5096 033450 112777 000000 144546  
 5097 033456 017704 144542  
 5098 033462 001011  
 5099 033464 017704 144536  
 5100 033470 042704 177770  
 5101 033474 022704 000002  
 5102 033500 001002  
 5103  
 5104 033502 000241  
 5105 033504 000401  
 5106  
 5107 033506 000261  
 5108 033510  
 5109 033510 012604  
 5110 033512 000207  
 5111  
 5112  
 5113

```

*****
:
:      SUBROUTINE - TINIT
:
:      THIS SUBROUTINE IS CALLED AT THE BEGINNING OF A TEST
:      TO DETERMINE IF A DEVICE RESET IS REQUIRED BEFORE
:      THE REST OF THE TEST IS EXECUTED.
:
:      INPUTS:          NONE
:
:      OUTPUTS:        IF A DEVICE RESET IS NOT REQUIRED
:                      THEN CARRY = 0
:
:                      IF A DEVICE RESET IS REQUIRED
:                      THEN CARRY = 1
:
:      CALLING SEQUENCE:
:                      JSR      PC,TINIT
:
*****
    
```

```

TINIT:
MOV      R4,-(SP)          ; SAVE R4
JSR      PC,CLINTR        ; ATTEMPT TO CLEAR PCSRO UPPER BYTE
MOVb     #ZERO,@PCSRO     ; INSURE LOWER BYTE CLEAR
MOVb     #ZERO,@PCSRO     ; MAY REQ. TWO WRITES
MOV      @PCSRO,R4        ; PCSRO = 0 ?
BNE      10$              ; NO, A RESET IS REQUIRED
MOV      @PCSR1,R4        ; PCSR1 -> R4
BIC      #SMASK,R4        ; MASK DELUA STATE
CMP      @READY,R4        ; STATE = READY ?
BNE      10$              ; NO, A RESET IS REQUIRED

CLC                          ; NO RESET REQUIRED, CLEAR CARRY
BR       20$

;
10$:    SEC                  ; A RESET IS REQUIRED, SET CARRY
20$:
MOV      (SP)+,R4         ; RESTORE R4
RTS      PC                ; AND RETURN
    
```



```
5126 .TITLE MISCELLANEOUS SECTIONS
5127 .SBTTL REPORT CODING SECTION
5148
5149
5150
5151 ;**
5152 ; THE REPORT CODING SECTION CONTAINS THE
5153 ; "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
5154 ;--
5155
5156 033514 BGNRPT L#RPT::
5157 033514
5158
5159 033514 EXIT RPT .WORD J#JMP
C'3514 000167 .WORD L10015-2-.
033516 000000
5160
5161
5162 .EVEN
5163
5164 033520 ENDRPT L10015:
033520 TRAP C#RPT
033520 104425
```

5166  
5167  
5168  
5169  
5170  
5171  
5172  
5173  
5174  
5175  
5176  
5177  
5178  
5179  
5180

033522  
033522  
033522 177777  
033524 177777  
033526 177777

.SBTTL PROTECTION TABLE

\*\*\*  
; THIS TABLE IS USED BY THE RUNTIME SERVICES  
; TO PROTECT THE LOAD MEDIA.  
\*\*\*

BGNPROT

L#PROT::

-1  
-1  
-1

;OFFSET INTO P-TABLE FOR CSR ADDRESS  
;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS  
;OFFSET INTO P-TABLE FOR DRIVE NUMBER

ENDPROT

```

5182          .SBTTL INITIALIZE SECTION
5183
5184          ;**
5185          ; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
5186          ; AT THE BEGINNING OF EACH PASS.
5187          ;--
5188
5189          BGNINIT
5190          033530          L$INIT::
5191          033530 005037 016610' CLR DNIFLG          ;INIT EXPECT DNI FLAG
5192          033534 012700 000036 REDEF #EF.CONTINUE ;WAS A CONTINUE COMMAND ENTERED?
5193          033542 104447          MOV #EF.CONTINUE,R0
5194          033542 103002          TRAP C$REFG
5195          033544 000137 034310' BNCOMPLETE 1$          ;NO, CONTINUE CHECK OF FLAGS
5196          033550 012700 000034 1$: JMP 60$          ;YES, LEAVE INIT CODE
5197          033554 104447          REDEF #EF.PWR          ;WAS THERE A POWER FAILURE?
5198          033556 103007          MOV #EF.PWR,R0
5199          BNCOMPLETE 3$          ;NO
5200          033560 012701 000150          TRAP C$REFG
5201          033564 005000          ; DELAY A PERIOD OF TIME (APPROX 25 SECS ) FOR SELF TEST TO FINISH
5202          033566 005300          ;
5203          033570 001376          MOV #150,R1          ;INIT OUTER LOOP
5204          033572 005301          CLR R0          ;INIT INNER LOOP
5205          033574 001374          2$: DEC R0
5206          033576 012700 000035          BNE 2$
5207          033602 104447          3$: REDEF #EF.NEW          ;NEW PASS ?
5208          033604 103065          MOV #EF.NEW,R0
5209          033606 012700 000040          TRAP C$REFG
5210          033612 104447          BNCOMPLETE 20$          ;NO
5211          033614 103054          BCC 20$
5212          033616 000005          REDEF #EF.START          ;START ?
5213          033620 005237 016610' 5208          MOV #EF.START,R0
5214          033624 012737 000001 016612'          TRAP C$REFG
5215          033632 012700 000114          BNCOMPLETE 5$          ;NO
5216          033636 104462          BCC 5$
5217          033640 010001          RESET          ;CLEAR THE WORLD
5218          033642 103412          INC DNIFLG          ;SET TO EXPECT RESULTING DNI
5219          033644 012746 021140'          MOV #1,FRSTIM          ;SET FIRST TIME FLAG
5220          033650 012746 000001          CLOCK L,R1          ;GET LINE CLOCK INFO
5221          033654 010600          MOV #L,R0
5222          033656 104417          TRAP C$CLCK
5223          033660 062706 000004          MOV R0,R1
5224          033664 000137 034302'          BCOMPLETE 4$          ;NO
5225          PRINTF #NOCLK          ;ERROR MESSAGE
5226          033644 012746 021140'          BCS 4$
5227          033650 012746 000001          MOV #NOCLK,-(SP)
5228          033654 010600          MOV #1,-(SP)
5229          033656 104417          MOV SP,R0
5230          033660 062706 000004          TRAP C$PNTF
5231          033664 000137 034302'          JMP 50$          ;CANNOT CONTINUE
5232          ADD #4,SP

```

```

5217 033670 012137 000246' 4$: MOV (R1)+,CLKCSR ;LINE CLOCK CSR
5218 033674 012102 MOV (R1)+,R2 ;GET CLOCK PRIORITY
5219 033676 072227 000005 ASH #5,R2
5220 033702 010237 000250' MOV R2,CLKBR
5221 033706 012137 000252' MOV (R1)+,CLKVEC ;VECTOR
5222 033712 012137 000254' MOV (R1)+,CLKFRE ;FREQUENCY
5223 033716 SETVEC CLKVEC,#CLKSRV,CLKBR ;SETUP CLOCK INTERRUPT VECTOR
    033716 013746 000250' MOV CLKBR,-(SP)
    033722 012746 034430' MOV #CLKSRV,-(SP)
    033726 013746 000252' MOV CLKVEC,-(SP)
    033732 012746 000003 MOV #3,-(SP)
    033736 104437 TRAP C$SVEC
    033740 062706 000010 ADD #10,SP

5224
5229
5230 033744 000402 BR 10$
5231 033746 005037 016612' 5$: CLR FRSTIM ;CLEAR FIRST TIME FLAG
5232 033752 012737 177777 000244' 10$: MOV #-1,UNIT ;YES, INITIALIZE UNIT NUMBER
5233 033760 005237 000244' 20$: INC UNIT ;SET UP FOR NEXT UNIT
5234 033764 023737 000244' 000012' CMP UNIT,L$UNIT ;TESTED ALL AVAILABLE UNITS?
5235 033772 003143 BGT 50$ ;YES, LEAVE
5236 033774 GPHARD UNIT,R1 ;GET P-TABLE POINTER FOR THIS UNIT
    033774 013700 000244' MOV UNIT,R0
    034000 104442 TRAP C$GPHRD
    034002 010001 MOV R0,R1

5237 034004 BNCOMPLETE 20$ ;THIS ONE IS NOT AVAILABLE
    034004 103365 BCC 20$

5238 034006 012137 000224' MCV (R1)+,PCSR0 ;SAVE PCSRO
5239 034012 012137 000240' MOV (R1)+,INTVEC ;SAVE VECTOR
5240 034016 013737 000224' 000234' MOV PCSRO,PCSR0UB ;SET UP ADDRESS OF UPPER BYTE OF PCSRO
5241 034024 062737 000001 000234' ADD #1,PCSR0UB
5242 034032 013737 000224' 000226' MOV PCSRO,PCSR1 ;SET UP PCSR1
5243 034040 062737 000002 000226' ADD #2,PCSR1
5244 034046 013737 000226' 000230' MOV PCSR1,PCSR2 ;SET UP PCSR2
5245 034054 062737 000002 000230' ADD #2,PCSR2
5246 034062 013737 000230' 000232' MOV PCSR2,PCSR3 ;SET UP PCSR3
5247 034070 062737 000002 000232' ADD #2,PCSR3

5248
5249 ;WAIT FOR DNI FROM PREVIOUS RESET IF APPROPRIATE
5250
5251 034076 005737 016610' TST DNIFLG ;EXPECTING DNI TO BE SET?
5252 034102 001502 BEQ 60$ ;NO, SKIP DNI HANDLING
5253 034104 004737 026550' JSR PC,CHKDNI ;WAIT FOR DNI
5254 034110 103057 BCC 30$ ;DNI?
5255 034112 017700 144106 MOV @PCSR0,R0 ;SAVE CONTENTS OF PCSRO
5256 034116 032700 001400 BIT #USCI!FATL,R0 ;UPROC. SUBSYSTEM FAILURE?
5257 034122 001011 BNE 22$ ;NO
5258 034124 PRINTF #M68FLD ;YES, ISSUE ERROR MESSAGE
    034124 012746 021202' MOV #M68FLD,(SP)
    034130 012746 000001 MOV #1,(SP)
    034134 010600 MOV SP,R0
    034136 104417 TRAP C$PNTF
    034140 062706 000004 ADD #4,SP

5259 034144 000456 BR 50$ ;CANNOT CONTINUE
5260 034146 032700 001000 22$: BIT #FATL,R0 ;DEVICE OR UNIBUS ERROR?
5261 034152 001411 BEQ 24$ ;NO
5262 034154 PRINTF #DEVUNI ;YES, REPORT ERROR
    
```

```

034154 012746 021310'
034160 012746 000001
034164 010600
034166 104417
034170 062706 000004
5263 034174 000442
5264 034176 032700 000400
5265 034202 001411
5266 034204
    034204 012746 021406'
    034210 012746 000001
    034214 010600
    034216 104417
    034220 062706 000004
5267 034224 000426
5268 034226
    034226 012746 021501'
    034232 012746 000001
    034236 010600
    034240 104417
    034242 062706 000004
5269 034246 000415
5270
5271
5272
5273 034250
5274 034250 004737 030264'
5275 034254 103011
5276 034256
    034256 012746 021575'
    034262 012746 000001
    034266 010600
    034270 104417
    034272 062706 000004
5277 034276 000401
5278 034300 000403
5279 034302 005037 016612'
5280 034306
    034306 104444
5281 034310 005037 016610'
5282 034314 005037 016620'
5283 034320
    034320 104421
5284 034322 032700 001000
5285 034326 001403
5286 034330 012737 000001 016620'
5287 034336
5288 034336
    034336 104411
5289
    
```

```

24$: BR 50$
    BIT #USCI,RO
    BEQ 26$
    PRINTF #NIUNIB
26$: BR 50$
    PRINTF #UNDFND
30$: JSR PC,CLRDN1
    BCC 40$
    PRINTF #DNICLR
40$: BR 50$
    BR 60$
50$: CLR FRSTIM
    DOCLN
60$: CLR DNIFLG
    CLR PRNTIT
    RFLAGS RO
70$: BIT #PNT,RO
    BEQ 70$
    MOV #1,PRNTIT
    ENDINIT
    
```

```

;CANNOT CONTINUE
;NI OR UNIBUS HALTED?
;NO
;YES, REPORT ERROR
;CANNOT CONTINUE
;REPORT BITS IN ILLEGAL STATE
;UNABLE TO CONTINUE
;YES, CLEAR IT
;CONTINUE IF DNI CLEARED
; ELSE ISSUE ERROR MESSAGE
; AND EXIT
;LEAVE
;CLEAR FIRST TIME FLAG
;ABORT PASS
;CLEAR EXPECT DNI FLAG
;CLEAR PRINT TEST ID FLAG
;READ FLAGS
;PRINT ENABLED?
;NO, DON'T SET PRINT TEST ID FLAG
;YES, SET FLAG
    
```

```

MOV #DEVUNI,-(SP)
MOV #1,-(SP)
MOV SP,RO
TRAP C$PNTF
ADD #4,SP
MOV #NIUNIB,-(SP)
MOV #1,-(SP)
MOV SP,RO
TRAP C$PNTF
ADD #4,SP
MOV #UNDFND,-(SP)
MOV #1,-(SP)
MOV SP,RO
TRAP C$PNTF
ADD #4,SP
MOV #DNICLR,-(SP)
MOV #1,-(SP)
MOV SP,RO
TRAP C$PNTF
ADD #4,SP
TRAP C$DCLN
TRAP C$RFLA
L10017: TRAP C$INIT
    
```

5291  
5292  
5293  
5294  
5295  
5296  
5297  
5298  
5299  
5300 034340  
034340  
5301  
5302  
5303 034340  
034340  
034340 104461

.SBTTL AUTODROP SECTION

; \*\*  
; THIS CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE CODE IF  
; THE "ADR" FLAG WAS SET. THE UNIT(S) UNDER TEST ARE CHECKED TO  
; SEE IF THEY WILL RESPOND. THOSE THAT DON'T ARE IMMEDIATELY  
; DROPPED FROM TESTING.  
; \*\*

BGNAUTO

L\$AUTO::

ENDAUTO

L10020: TRAP C\$AUTO

5305  
5306  
5307  
5308  
5309  
5310  
5311  
5312 034342  
034342  
5313  
5314  
5315 034342  
034342 104432  
034344 000002  
5316  
5317  
5318  
5319  
5320 034346  
034346  
034346 104412

.SBTTL CLEANUP CODING SECTION  
; \*\*  
; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED  
; AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED.  
; --

BGNCLN

L\$CLEAN::

EXIT CLN

TRAP C\$EXIT  
.WORD L10021--

.EVEN

ENDCLN

L10021:  
TRAP C\$CLEAN

```
5322 .SBTTL DROP UNIT SECTION
5323
5324 ;++
5325 ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
5326 ; TO NO LONGER BE TESTED.
5327 ;--
5328
5329 034350 BGNDU
5330 034350 L#DU::
5331
5332 034350 EXIT DU
5333 034350 000167 .WORD J#JMP
5334 034352 000000 .WORD L10022-2
5335
5336 .EVEN
5337 034354 ENDDU
5338 034354 L10022:
5339 034354 104453 TRAP C#DU
```



```
5339          .SBTTL  ADD UNIT SECTION
5340
5341          ;**
5342          ; THE ADD-UNIT SECTION CONTAINS ANY CODE THE PROGRAMMER WISHES
5343          ; TO BE EXECUTED IN CONJUNCTION WITH THE ADDING OF A UNIT BACK
5344          ; TO THE TEST CYCLE.
5345          ;--
5346
5347 034356      BGNAU
5348          034356
5349
5350 034356      EXIT  AU
5351          034356 000167
5352          034360 000000
5353
5354          .EVEN
5355 034362      ENDAU
5356          034362
5357          034362 104452

L$AU::
        .WORD  J$JMP
        .WORD  L10023-2-

L10023: TRAP  C$AU
```

```
5357 .TITLE GLOBAL INTERRUPT SERVICE ROUTINES
5358
5359 .SBTTL ISRNXM - NON-EXISTANT MEMORY INTERRUPT SERVICE ROUTINE
5360
5361 ;*****
5362 ;
5363 ; FUNCTIONAL DESCRIPTION:
5364 ;
5365 ; THIS ROUTINE IS ASSIGNED TO VECTOR 4 BY THE ACCESS TESTS.
5366 ; WHEN AN ACCESS IS ATTEMPTED ON NON-EXISTENT MEMORY
5367 ; THE NEXMEM FLAG IS SET.
5368 ;
5369 ;*****
5370
5371 034364 BGNSRV ISRNXM ISRNXM::
034364
5372 034364 012737 000001 016604' MOV #1,NEXMEM ;SET NXM FLAG
5373 034372 ENDSRV L10024:
034372 000002 RTI
5374
```

```
5376
5377 .SBTTL ISRDN1 - DNI INTERRUPT SERVICE ROUTINE
5378
5379 ;*****
5380 ;
5381 ; FUNCTIONAL DESCRIPTION:
5382 ;
5383 ; THIS ROUTINE IS ASSIGNED TO THE DELUA'S INTERRUPT VECTOR BY
5384 ; TEST 10.
5385 ; WHEN AN INTERRUPT OCCURS THE DNIFLG FLAG IS SET IF DNI IS SET.
5386 ;
5387 ;*****
5388
5389 034374 BGNSRV ISRDN1
5390 034374 ISRDN1::
5391 034374 010446 MOV R4,-(SP) ; SAVE R4
5392 034376 005037 016610' CLR DNIFLG ; INSURE DNI FLAG IS CLEAR
5393 034402 005004 CLR R4 ; INSURE R4 IS CLEAR
5394 034404 017704 143614 MOV @PCSR0,R4 ; PCSRO -> R4
5395 034410 032704 004000 BIT #DNI,R4 ; DNI SET?
5396 034414 001403 BEQ 10$ ; NO, EXIT
5397 034416 012737 000001 016610' MOV #1,DNIFLG ; YES, SET DNIFLG FLAG
5398 034424 012604 10$: MOV (SP)+,R4 ; RESTORE R4
5399
5400 034426 ENDSRV
5401 034426 L10025:
5402 034426 000002 RTI
```

```
5402
5403 ;*****
5404 ;
5405 ;FUNCTIONAL DESCRIPTION:
5406 ;
5407 ; THIS ROUTINE COUNTS A PRESET NUMBER OF CLOCK TICKS THEN IT
5408 ; TURNS THE CLOCK OFF
5409 ;
5410 ;INPUTS: METER
5411 ;
5412 ;OUTPUTS:METER
5413 ;
5414 ;ROUTINES CALLED: NONE
5415 ;
5416 ;*****
5417
5418 034430 BGNSRV CLKSRV
5419 034430
5419 034430 005737 016602' TST METER ;HAS THE METER EXPIRED? CLKSRV::
5420 034434 001402 BEQ 20$ ;YES, STOP COUNTING
5421 034436 005337 016602' DEC METER ;COUNT TICKS
5422 034442 20$:
5423 034442 ENDSRV
034442 000002 L10026: RTI
```

```

5426 .TITLE HARDWARE TESTS
5437
5459 .SBTTL TEST 1: PCSRO READ ACCESS TEST
5460
5461 ;*****
5462 ;
5463 ; THIS TEST WILL VERIFY THAT A DEVICE IS PRESENT AT THE PCSRO
5464 ; UNIBUS ADDRESS SPECIFIED.
5465 ;
5466 ; TEST SEQUENCE:
5467 ; 1. READ PCSRO
5468 ;*****
5469
5470
5471 034444 BGNTST
5472 034444 T1::
5473 034444
5474 034444 012704 034570'
5475 034450 004737 032734'
5476 034444 012704 034570' ;GET POINTER TO TEST NAME MESSAGE
5477 034450 004737 032734' ;PRINT TEST NUMBER AND NAME
5478 ;
5479 ; END OF MACRO EXPANSION OF 'PNTMAC'
5480 034454 SETVEC #4,#ISRNXM,#PRI07 ; SET UP TIMEOUT TRAP VECTOR
5481 034454 012746 000340 MOV #PRI07,-(SP)
5482 034460 012746 034364' MOV #ISRNXM,-(SP)
5483 034464 012746 000004 MOV #4,-(SP)
5484 034470 012746 000003 MOV #3,-(SP)
5485 034474 104437 TRAP C#SVEC
5486 034476 062706 000010 ADD #10,SP
5487 034502 005037 016604' CLR NEXMEM ; CLEAR NXM TIMEOUT FLAG
5488 034506 005002 CLR R2 ; R2 = WHICH PCSR IS BEING TESTED
5489 034510 012777 004000 143506 MOV #4000,#PCSR0 ; DOES PCSR EXIST?
5490 034516 005737 016604' TST NEXMEM
5491 034522 001413 BEQ 10# ; YES
5492 034524 CLRVEC #4
5493 034524 012700 000004 MOV #4,R0
5494 034530 104436 TRAP C#CVEC
5495 034532 ERRDF 001,ERR001,MSG001 ; NO. PRINT DEVICE FATAL ERROR MESSAGE
5496 034532 104455 TRAP C#ERDF
5497 034534 000001 .WORD 1
5498 034536 022660' .WORD ERR001
5499 034540 021664' .WORD MSG001
5500 034542 DODU UNIT ; DROP UNIT
5501 034542 013700 000244' MOV UNIT,R0
5502 034546 104451 TRAP C#DODU
5503 034550 DOCLN
5504 034550 104444 TRAP C#DCLN
5505 034552 10# CLRVEC #4
5506 034552 012700 000004 MOV #4,R0
5507 034556 104436 TRAP C#CVEC
5508 034560 004737 030346' JSR PC,CLINTR ;INSURE DELUA INTR ARE CLEAR
5509 034564 EXIT TST
5510 034564 104432 TRAP C#EXIT
  
```

034566 000034

.WORD L10027-

5489

5490

;LOCAL TEST MESSAGE

5491

5492 034570 104 105 114 T01ID: .ASCIZ 'DELUA PCSRO READ ACCESS '

034573 125 101 040

034576 120 103 123

034601 122 060 040

034604 122 105 101

034607 104 040 101

034612 103 103 105

034615 123 123 040

034620 000

5493

.EVEN

5494

5495 034622

ENDTST

L10027:

034622

TRAP

C#ETST

034622 104401

5496

```

5498 .SBTTL TEST 2: PCSR1 READ ACCESS TEST
5499
5500 :*****
5501 :
5502 : THIS TEST WILL VERIFY THAT A DEVICE IS PRESENT AT THE PCSR1
5503 : UNIBUS ADDRESS SPECIFIED.
5504 :
5505 : TEST SEQUENCE:
5506 : 1. READ PCSR1
5507 :
5508 :*****
5509
5510 034624 BGNTST
5511 034624 T2::
5512 034624 PNTMAC T02ID
034624 012704 034756' MOV #T02ID,R4 ;GET POINTER TO TEST NAME MESSAGE
034630 004737 032734' JSR PC,PNTID ;PRINT TEST NUMBER AND NAME
:
: END OF MACRO EXPANSION OF 'PNTMAC'
5513 034634 012777 004000 143362 MOV #4000,@PCSR0 ; INSURE DNI CLEAR
5514 034642 SETVEC #4,@ISRNXM,@PRI07 ; SET UP TIMEOUT TRAP VECTOR
034642 012746 000340 MOV #PRI07,-(SP)
034646 012746 034364' MOV #ISRNXM,-(SP)
034652 012746 000004 MOV #4,-(SP)
034656 012746 000003 MOV #3,-(SP)
034662 104437 TRAP C$SVEC
034664 062706 000010 ADD #10,SP
5515 034670 005037 016604' CLR NEXMEM ; CLEAR NXM TIMEOUT FLAG
5516 034674 012702 000001 MOV #1,R2 ; R2 = WHICH PCSR IS BEING TESTED
5517 034700 017701 143322 MOV @PCSR1,R1 ; DOES PCSR EXIST?
5518 034704 005737 016604' TST NEXMEM
5519 034710 001413 BEQ 10$ ; YES
5520 034712 CLRVEC #4
034712 012700 000004 MOV #4,R0
034716 104436 TRAP C$CVEC
5521 034720 ERRDF 002,ERR001,MSG001 ; NO, PRINT DEVICE FATAL FRROR MESSAGE
034720 104455 TRAP C$ERDF
034722 000002 .WORD 2
034724 022660' .WORD ERR001
034726 021664' .WORD MSG001
5522 034730 DODU UNIT ; DROP UNIT
034730 013700 000244' MOV UNIT,R0
034734 104451 TRAP C$DODU
5523 034736 DOCLN ; AND ABORT PASS
034736 104444 TRAP C$DOCLN
5524 034740 10$: CLRVEC #4
034740 012700 000004 MOV #4,R0
034744 104436 TRAP C$CVEC
5525 034746 004737 030346' JSR PC,CLINTR ;INSURE PCSRO INTR ARE CLEARED
5526
5527 034752 EXIT TST
034752 104432 TRAP C$EXIT
034754 000034 .WORD L10030-.

```

```
5529                                     ;LOCAL TEST MESSAGE
5530
5531 034756      104      105      114 T02ID:.ASCIZ 'DELUA PCSR1 READ ACCESS '
      034761      125      101      040
      034764      120      103      123
      034767      122      061      040
      034772      122      105      101
      034775      104      040      101
      035000      103      103      105
      035003      123      123      040
      035006      000
5532                                     .EVEN
5533
5534 035010      ENDTST
      035010
      035010 104401
```

L10030: TRAP C#ETST



5536  
 5537  
 5538  
 5539  
 5540  
 5541  
 5542  
 5543  
 5544  
 5545  
 5546  
 5547  
 5548

.SBTTL TEST 3: PCSR1 DELUA ID BIT TEST

```

*****
:
: THIS TEST VERIFIES THAT BIT 06, AND NO OTHER BITS IN THE
: PCSR1 DEVICE ID FIELD IS SET.
:
: TEST SEQUENCE:
:     1. READ PCSR1
*****
    
```

5549 035012  
 035012  
 5550  
 5551 035012

BGNTST

T3::

PNTMAC T03ID

035012 012704 035172'  
 035016 004737 032734'

```

MOV    #T03ID,R4      ;GET POINTER TO TEST NAME MESSAGE
JSR    PC,PNTID       ;PRINT TEST NUMBER AND NAME
    
```

: END OF MACRO EXPANSION OF 'PNTMAC'

5552  
 5553 035022 004737 033434'  
 5554 035026 103034  
 5555 035030 012777 004100 143166  
 5556 035036 112777 000140 143160  
 5557 035044 004737 027736'  
 5558 035050 103010  
 5559 035052 004737 026652'  
 5560 035056  
 035056 104456  
 035060 000003  
 035062 023030'  
 035064 021736'

```

JSR    PC,TINIT      ;IS A DEVICE RESET NEEDED?
BCC    20$           ;NO
MOV    #DNI+INTE,@PCSR0 ;PRECONDITION INTR ENABLE
MOVB   #INTE+RSET,@PCSR0 ;RESET DELUA
JSR    PC,CKDNI      ;DNI?
BCC    10$           ;YES
JSR    PC,CHKFTL     ;FATAL BIT SET?
ERRHRD 003,ERR006,MSG003 ;NO, REPORT ERROR
    
```

```

TRAP   C$ERHRD
.WORD  3
.WORD  ERR006
.WORD  MSG003
    
```

5561 035066  
 035066 104410  
 035070 000126

ESCAPE TST ; AND ABORT TEST

```

TRAP   C$ESCAPE
.WORD  L10031-.
    
```

5562 035072  
 5563 035072 004737 030264'  
 5564 035076 103010  
 5565 035100 004737 026652'  
 5566 035104  
 035104 104456  
 035106 000004  
 035110 023030'  
 035112 021736'

10\$:

```

JSR    PC,CLRDN1     ;WRITE 1 TO CLEAR DNI
BCC    20$           ;OK TO CONTINUE
JSR    PC,CHKFTL     ;FATAL BIT SET?
ERRHRD 004,ERR006,MSG003 ;NO, REPORT ERROR
    
```

```

TRAP   C$ERHRD
.WORD  4
.WORD  ERR006
.WORD  MSG003
    
```

5567 035114  
 035114 104410  
 035116 000100

ESCAPE TST ; AND ABORT TEST

```

TRAP   C$ESCAPE
.WORD  L10031 .
    
```

5568  
 5569  
 5570 035120  
 5571 035120 017701 '3102  
 5572 035124 142701 000217  
 5573 035130 122701 000020

20\$:

```

MOV    @PCSR1,R1     ;GET CONTENTS OF PCSR1
BICB   #217,R1       ;CLEAR UNWANTED BITS
CMPB   #20,R1        ;ONLY BIT4 SET?
    
```

```

5574 035134 001412          BEQ      30$          ;YES, SKIP ERROR REPORT
5575 035136 004737 026652' JSR      PC,CHKFTL    ;FATAL BIT SET?
5576 035142          ERRDF    005,ERR040
      035142 104455          TRAP    C$ERDF
      035144 000005          .WORD  5
      035146 025676'        .WORD  ERR040
      035150 000000          .WORD  0
5577 035152          DODU     UNIT        ;ILLEGAL ID, DROP UNIT
      035152 013700 000244' MOV     UNIT,RO
      035156 104451          TRAP    C$DODU
5578
5579 035160          DOCLN                    ; AND ABORT PASS
      035160 104444          TRAP    C$DCLN
5580 035162          30$:
5581 035162 004737 030346' JSR      PC,CLINTR    ;INSURE DELUA INTR BIT CLEAR
5582
5583 035166          EXIT     TST
      035166 104432          TRAP    C$EXIT
      035170 000026          .WORD  L10031-.
5584
5585          ;LOCAL TEST MESSAGE
5586
5587 035172          104      105      114  T03ID: .ASCIZ 'DELUA PCSR1 ID BIT '
      035175          125      101      040
      035200          120      103      123
      035203          122      061      040
      035206          111      104      040
      035211          102      111      124
      035214          040      000
5588          .EVEN
5589
5590 035216          ENDTST
      035216          L10031:
      035216 104401          TRAP    C$ETST
5591
  
```

```

5593          .SBTTL TEST 4: PCSR2 READ ACCESS TEST
5594
5595          ;*****
5596          ;
5597          ; THIS TEST WILL VERIFY THAT A DEVICE IS PRESENT AT THE PCSR2
5598          ; UNIBUS ADDRESS SPECIFIED.
5599          ;
5600          ; TEST SEQUENCE:
5601          ;     1. READ PCSR2
5602          ;*****
5603
5604
5605 035220      BGNTST
5606           035220
5607 035220
5608           035220 012704 035352'
5609           035224 004737 032734'
5610           ;
5611           ; END OF MACRO EXPANSION OF 'PNTMAC'
5612
5613 035230      MOV      #4000, @PCSR0          ; INSURE DNI CLEAR
5614 035236      SETVEC  #4, @ISRNXM, @PRI07    ; SET UP TIMEOUT TRAP VECTOR
5615           035236 012746 000340          MOV      #PRI07, -(SP)
5616           035242 012746 034364'          MOV      #ISRNXM, -(SP)
5617           035246 012746 000004          MOV      #4, -(SP)
5618           035252 012746 000003          MOV      #3, -(SP)
5619           035256 104437          TRAP    C$SVEC
5620           035260 062706 000010          ADD     #10, SP
5621 035264      CLR      NEXMEM                ; CLEAR NXM TIMEOUT FLAG
5622 035270      MOV      #2, R2                  ; R2 = WHICH PCSR IS BEING TESTED
5623 035274      MOV      @PCSR2, R1            ; DOES PCSR EXIST?
5624 035300      TST     NEXMEM
5625 035304      BEQ     10$
5626 035306      CLRVEC #4                      ; YES
5627           035306 012700 000004          MOV      #4, R0
5628           035312 104436          TRAP    C$CVEC
5629 035314      ERRDF  006, ERR001, MSG001     ; NO, PRINT DEVICE FATAL ERROR MESSAGE
5630           035314 104455          TRAP    C$ERDF
5631           035316 000006          .WORD   6
5632           035320 022660'          .WORD   ERR001
5633           035322 021664'          .WORD   MSG001
5634 035324      DODU   UNIT                    ; DROP UNIT
5635           035324 013700 000244'          MOV      UNIT, R0
5636           035330 104451          TRAP    C$DODU
5637 035332      DOCLN
5638           035332 104444          TRAP    C$DCLN
5639 035334      10$: CLRVEC #4
5640           035334 012700 000004          MOV      #4, R0
5641           035340 104436          TRAP    C$CVEC
5642 035342      JSR    PC, CLINTR              ; INSURE DELUA INTR BITS CLEAR
5643 035346      EXIT   TST
5644           035346 104432          TRAP    C$EXIT
5645           035350 000034          .WORD   L10032
    
```

```
5625 ;LOCAL TEST MESSAGE
5626
5627 035352 104 105 114 T04ID: .ASCIZ 'DELUA PCSR2 READ ACCESS '
      035355 125 101 040
      035360 120 103 123
      035363 122 062 040
      035366 122 105 101
      035371 104 040 101
      035374 103 103 105
      035377 123 123 040
      035402 000
```

5628 .EVEN

5629  
5630 035404 ENDTST

035404  
035404 104401

L10032: TRAP C\$ETST

```

5632 .SBTTL TEST 5: PCSR3 READ ACCESS TEST
5633
5634 ;*****
5635 ;
5636 ; THIS TEST WILL VERIFY THAT A DEVICE IS PRESENT AT THE PCSR3
5637 ; UNIBUS ADDRESS SPECIFIED.
5638 ;
5639 ; TEST SEQUENCE:
5640 ; 1. READ PCSR3
5641 ;*****
5642 ;
5643 035406 BGNTST
5644 035406 TS::
5645 035406 PNTMAC T05ID
035406 012704 035540' MOV #T05ID,R4 ;GET POINTER TO TEST NAME MESSAGE
035412 004737 032734' JSR PC,PNTID ;PRINT TEST NUMBER AND NAME
; END OF MACRO EXPANSION OF 'PNTMAC'
5646 035416 012777 004000 142600 MOV #4000,@PCSR0 ; INSURE DNI CLEAR
5647 035424 SETVEC #4,@ISRNXM,@PRIO7 ; SET UP TIMEOUT TRAP VECTOR
035424 012746 000340 MOV #PRIO7,-(SP)
035430 012746 034364' MOV #ISRNXM,-(SP)
035434 012746 000004 MOV #4,-(SP)
035440 012746 000003 MOV #3,-(SP)
035444 104437 TRAP C$SVEC
035446 062706 000010 ADD #10,SP
5648 035452 005037 016604' CLR NEXMEM ; CLEAR NXM TIMEOUT FLAG
5649 035456 012702 000003 MOV #3,R2 ; R2 = WHICH PCSR IS BEING TESTED
5650 035462 017701 142544 MOV @PCSR3,R1 ; DOES PCSR EXIST?
5651 035466 005737 016604' TST NEXMEM
5652 035472 001413 BEQ 10$ ; YES
5653 035474 CLRVEC #4
035474 012700 000004 MOV #4,R0
035500 104436 TRAP C$CVEC
5654 035502 ERRDF 007,ERR001,MSG001 ; NO. PRINT DEVICE FATAL ERROR MESSAGE
035502 104455 TRAP C$ERDF
035504 000007 .WORD 7
035506 022660' .WORD ERR001
035510 021664' .WORD MSG001
5655 035512 DODU UNIT ; DROP UNIT
035512 013700 000244' MOV UNIT,R0
035516 104451 TRAP C$DODU
5656 035520 DOCLN ; AND ABORT PASS
035520 104444 TRAP C$DCLN
5657
5658 035522 10$: CLRVEC #4
035522 012700 000004 MOV #4,R0
035526 104436 TRAP C$CVEC
5659 035530 004737 030345' JSR PC,CLINTR ; INSURE DELUA INTR BITS DISABLED
5660
5661 035534 EXIT TST
035534 104432 TRAP C$EXIT
035536 000034 .WORD L10033-
    
```

5663 ;LOCAL TEST MESSAGE  
5664  
5665 035540 104 105 114 TOSID: .ASCIZ 'DELUA PCSR3 READ ACCESS '  
035543 125 101 040  
035546 120 103 123  
035551 122 063 040  
035554 122 105 101  
035557 104 040 101  
035562 103 103 105  
035565 123 123 040  
035570 000

5666 .EVEN

5667  
5668 035572 ENDTST

L10033: TRAP C#ETST

035572 104401

5670  
5671  
5672  
5673  
5674  
5675  
5676  
5677  
5678  
5679  
5680  
5681  
5682  
5683  
5684  
5685  
5686  
5687  
5688  
5689  
5690  
  
5691  
5692  
5693  
5694  
5695  
5696  
5697  
5698  
5699  
  
5700  
  
5701  
5702  
5703  
5704  
5705  
  
5706  
  
5707

.SBTTL TEST 6: PCSR2 STATIC BIT TEST

```
*****  
: THIS TEST WILL CHECK PCSR2 FOR ALL SA0 AND SA1 ERRORS.  
: THE HOST WILL WRITE PATTERNS TO PCSR2 AND READ THEM  
: BACK TO VERIFY.  
:  
: NOTE: PCSR2 BIT00 SHOULD ALWAYS BE A ZERO.  
: THIS BIT WILL BE MASKED BEFORE DOING THE COMPARE.  
:  
: TEST SEQUENCE:  
: 1. WRITE PATTERN TO PCSR2  
: 2. COMPARE MASKED PATTERN WITH PCSR2 CONTENTS  
: 3. REPEAT STEPS 1 AND 2 FOR ALL PATTERNS  
:*****
```

BGNTST

T6::

PNTMAC T06ID

035574 012704 035760'  
035600 004737 032734'

MOV #T06ID,R4 ;GET POINTER TO TEST NAME MESSAGE  
JSR PC,PNTID ;PRINT TEST NUMBER AND NAME

END OF MACRO EXPANSION OF 'PNTMAC'

JSR PC,TINIT ;IS A DEVICE RESET NEEDED?  
BCC 25\$ ;NO  
MOV #DNI+INTE,@PCSR0 ;PRE-CONDITION INTR ENABLE  
MOVB #INTE+RSET,@PCSR0 ;RESET DELUA  
JSR PC,CKDNI ;DNI?  
BCC 15\$ ;YES  
JSR PC,CHKFTL ;FATAL BIT SET?  
ERRHRD 008.,ERR006,MSG003 ;NO, REPORT ERROR

TRAP C\$ERHRD  
.WORD 8  
.WORD ERR006  
.WORD MSG003

ESCAPE TST ; AND ABORT TEST

TRAP C\$ESCAPE  
.WORD L10034 .

15\$:

JSR PC,CLRDN1 ;WRITE 1 TO CLEAR DNI BIT  
BCC 25\$ ;NO  
JSR PC,CHKFTL ;FATAL BIT SET?  
ERRHRD 010.,ERR006,MSG003 ;YES, REPORT ERROR

TRAP C\$ERHRD  
.WORD 10  
.WORD ERR006  
.WORD MSG003

ESCAPE TST ; AND ABORT TEST

TRAP C\$ESCAPE  
.WORD L10034-.

```

5708 035702          25$:
5709
5710 035702 012701 016626'      MOV    #PATRN1,R1      ; GET ADDRESS OF DATA PATTERNS
5711 035706 012705 000004      MOV    #4,R5           ; COUNT 4 PATTERNS (PASSES)
5712 035712 012103          30$: MOV    (R1)+,R3       ; DATA PATTERN -> R3
5713
5714
5715 035714 010377 142310      MOV    R3,@PCSR2      ; DATA PATTERN -> PCSR2
5716 035720 017704 142304      MOV    @PCSR2,R4      ; READ PCSR2
5717 035724 020304          CMP    R3,R4          ; DATA COMPARE?
5718 035726 001406          BEQ    50$            ; YES, CONTINUE
5719 035730 004737 026652'      JSR    PC,CHKFTL      ; FATAL BIT SET?
5720 035734          ERRHRD 011.,ERR002,MSG002 ; NO, REPORT ERROR
      TRAP    C$ERHRD
      .WORD  11
      .WORD  ERR002
      .WORD  MSG002
      035734 104456
      035736 000013
      035740 022710'
      035742 021710'
5721 035744          50$:
5722 035744 005305      DEC    R5             ; DONE?
5723 035746 001361      BNE   30$            ; NO
5724
5725 035750 004737 030346'      JSR    PC,CLINTR     ; INSURE DELUA INTR BITS CLEAR
5726
5727 035754          EXIT   TST
      TRAP    C$EXIT
      .WORD  L10034-.
      035754 104432
      035756 000032
5728
5729          ;LOCAL TEST MESSAGE
5730
5731 035760          104    105    114  T06ID: .ASCIZ 'DELUA PCSR2 STATIC BIT '
      035763          125    101    040
      035766          120    103    123
      035771          122    062    040
      035774          123    124    101
      035777          124    111    103
      036002          040    102    111
      036005          124    040    000
5732          .EVEN
5733
5734 036010          ENDTST
      L10034:
      036010          TRAP    C$ETST
      036010 104401
  
```



5736  
 5737  
 5738  
 5739  
 5740  
 5741  
 5742  
 5743  
 5744  
 5745  
 5746  
 5747  
 5748  
 5749  
 5750  
 5751  
 5752  
 5753

.SBTTL TEST 7: PCSR3 STATIC BIT TEST

```

*****
;
; THIS TEST WILL CHECK PCSR3 FOR ALL SA0 AND SA1 ERRORS.
; THE HOST WILL WRITE PATTERNS TO PCSR3 AND READ THEM
; BACK TO VERIFY.
;
; NOTE: PCSR3 BIT02 THRU BIT15 SHOULD ALWAYS BE A ZERO.
; THESE BITS WILL BE MASKED BEFORE DOING THE COMPARE.
;
; TEST SEQUENCE:
; 1. WRITE PATTERN TO PCSR3
; 2. COMPARE MASKED PATTERN WITH PCSR3 CONTENTS
; 3. REPEAT STEPS 1 AND 2 FOR ALL PATTERNS
*****
    
```

5754 036012  
 036012  
 5755  
 5756 036012

BGNTST

T7::

PNTMAC T07ID

036012 012704 036176'  
 036016 004737 032734'

```

MOV    @T07ID,R4      ;GET POINTER TO TEST NAME MESSAGE
JSR    PC,PNTID       ;PRINT TEST NUMBER AND NAME
    
```

END OF MACRO EXPANSION OF 'PNTMAC'

5757  
 5758 036022 004737 033434'  
 5759 036026 103034  
 5760 036030 012777 004100 142166  
 5761 036036 112777 000140 142160  
 5762 036044 004737 027736'  
 5763 036050 103010  
 5764 036052 004737 026652'  
 5765 036056  
       036056 104456  
       036060 000014  
       036062 023030'  
       036064 021736'  
 5766 036066  
       036066 104410  
       036070 000136

```

JSR    PC,TINIT      ;IS A DEVICE RESET NEEDED?
BCC    20$           ;NO
MOV    @DNI+INTE,@PCSR0 ;PRE-CONDITION INTR ENABLE
MOVB   @INTE+RSET,@PCSR0 ;YES, RESET DELUA
JSR    PC,CKDNI      ;DNI SET?
BCC    10$           ;YES
JSR    PC,CHKFTL     ;FATAL BIT SET?
ERRHRD 012.,ERR006,MSG003 ;NO, REPORT ERROR
    
```

TRAP C\$ERHRD  
 .WORD 12  
 .WORD ERR006  
 .WORD MSG003

ESCAPE TST ; AND ABORT TEST

TRAP C\$ESCAPE  
 .WORD L10035--

5767 036072  
 5768 036072 004737 030264'  
 5769 036076 103010  
 5770 036100 004737 026652'  
 5771 036104  
       036104 104456  
       036106 000015  
       036110 023030'  
       036112 021736'  
 5772 036114  
       036114 104410  
       036116 000110

10\$:

```

JSR    PC,CLRDN1     ;WRITE 1 TO CLEAR DNI BIT
BCC    20$           ;CLEARED OK
JSR    PC,CHKFTL     ;FATAL BIT SET?
ERRHRD 013.,ERR006,MSG003 ;NO, REPORT ERROR
    
```

TRAP C\$ERHRD  
 .WORD 13  
 .WORD ERR006  
 .WORD MSG003

ESCAPE TST ; AND ABORT TEST

TRAP C\$ESCAPE  
 .WORD L10035--

5773

```

5774 036120          20$:
5775 036120 012701 016626'   MOV    @PATRN1,R1      ; GET ADDRESS OF DATA PATTERNS
5776 036124 012705 000004'   MOV    @4,R5          ; COUNT 4 PATTERNS (PASSES)
5777 036130 012103          40$:   MOV    (R1)+,R5       ; DATA PATTERN -> R3
5778
5779 036132 010377 142074   MOV    R3,@PCSR3     ; DATA PATTERN -> PCSR3
5780 036136 017704 142070   MOV    @PCSR3,R4     ; READ PCSR3
5781 036142 020304          CMP    R3,R4         ; DATA COMPARE?
5782 036144 001406          BEQ    50$           ; YES, CONTINUE
5783 036146 004737 026652'   JSR    PC,CHKFTL     ; FATAL ERROR BIT SET?
5784 036152          ERRHRD 014.,ERR003,MSG002 ; NO, REPORT ERROR
          036152 104456          TRAP   C$ERHRD
          036154 000016          .WORD 14
          036156 022746'        .WORD ERR003
          036160 021710'        .WORD MSG002

5785
5786 036162          50$:
5787 036162 005305          DEC    R5             ; DONE?
5788 036164 001361          BNE   40$           ; NO
5789
5790
5791 036166 004737 030346'   JSR    PC,CLINTR     ; INSURE INTR BITS ARE CLEAR
5792
5793 036172          EXIT   TST
          036172 104432          TRAP   C$EXIT
          036174 000032          .WORD L10035-.

5794
5795          ;LOCAL TEST MESSAGE
5796
5797 036176          104    105    114  T07ID: .ASCIZ 'DELUA PCSR3 STATIC BIT '
          036201          125    101    040
          036204          120    103    123
          036207          122    063    040
          036212          123    124    101
          036215          124    111    103
          036220          040    102    111
          036223          124    040    000

5798          .EVEN
5799
5800 036226          ENDTST
          036226          L10035:
          036226 104401          TRAP   C$ETST
    
```

TEST 8: SELF TEST

5802  
5803  
5804  
5805  
5806  
5807  
5808  
5809  
5810  
5811  
5812  
5813  
5814  
5815  
5816  
5817

.SBTTL TEST 8: 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. ISSUE THE SELF TEST PORT COMMAND
;   2. WAIT FOR DNI
;   3. CHECK LITE BITE REGISTER FOR SUCCESSFUL SELF TEST
;   4. WRITE ONE TO CLEAR DNI
;
*****
    
```

5818 036230  
5819 036230  
5820 036230

BGNTST

T8::

PNTMAC TO8ID

036230 012704 036534'  
036234 004737 032734'

```

MOV #TO8ID,R4 ;GET POINTER TO TEST NAME MESSAGE
JSR PC,PNTID ;PRINT TEST NUMBER AND NAME
    
```

; END OF MACRO EXPANSION OF 'PNTMAC'

5821 036240 004737 033434'  
5822 036244 103034  
5823 036246 012777 004100 141750  
5824 036254 112777 000140 141742  
5825 036262 004737 027736'  
5826 036266 103010  
5827 036270 004737 026652'  
5828 036274  
036274 104456  
036276 000017  
036300 023030'  
036302 021736'  
5829 036304  
036304 104410  
036306 000246

```

JSR PC,TINIT ; IS A DEVICE RESET NEEDED?
BCC 25# ; NO
MOV #DNI+INTE,@PCSR0 ; SET INTERRUPT ENABLE
MOVB #INTE!RSET,@PCSR0 ; YES, RESET DELUA
JSR PC,CKDNI ; DNI ?
BCC 20# ; YES
JSR PC,CHKFTL ; FATL BIT SET?
ERRHRD 015.,ERR006,MSG003 ; NO, REPORT ERROR
    
```

TRAP C#ERRHRD  
.WORD 15  
.WORD ERR006  
.WORD MSG003

ESCAPE TST ; AND ABORT TEST

TRAP C#ESCAPE  
.WORD L10036--

5830  
5831 036310 004737 030264'  
5832  
5833 036314 103010  
5834 036316 004737 026652'  
5835 036322  
036322 104456  
036324 000020  
036326 023030'  
036330 021736'  
5836 036332  
036332 104410  
036334 000220

```

; 20#: JSR PC,CLRDN1 ; WRITE ONE TO CLEAR DNI
; ERROR
BCC 25# ; NO
JSR PC,CHKFTL ; FATL BIT SET?
ERRHRD 016.,ERR006,MSG003 ; YES, REPORT ERROR
    
```

TRAP C#ERRHRD  
.WORD 16  
.WORD ERR006  
.WORD MSG003

ESCAPE TST ; AND ABORT

TRAP C#ESCAPE  
.WORD L10036--

5837  
5838 036336  
5839 036336 012777 004100 141660

```

; 25#: MOV #DNI+INTE,@PCSR0 ; PRE-CONDITION INTR ENABLE
    
```

TEST 8: SELF TEST

```

5840 036344 112777 000103 141652      MOVB  @INTE+SLFT,@PCSR0      ; RUN SELF TEST
5841 036352 004737 027736'          JSR   PC,CKDNI              ; SELF TEST COMPLETED OK?
5842 036356 103025                    BCC   35$                  ; YES, CHECK RESULTS
5843
5844                                     ;FIND OUT WHY SELF TEST DID NOT COMPLETE
5845
5846 036360 013700 016514'          MOV   EPCSR0,R0            ; NO, GET CONTENTS OF PCSRO
5847 036364 042700 172377          BIC   @STATEM,R0          ; MASK UNWANTED BITS
5848 036370 005700                    TST   R0                  ; 68000 SUBSYSTEM FAULT
5849 036372 001006                    BNE   30$                  ; NO
5850 036374                    ERRHRD 017.,ERR011,MSG011   ; YES, REPORT ERROR
                                     TRAP   C$ERHRD
                                     .WORD  17
                                     .WORD  ERR011
                                     .WORD  MSG011
5851 036404                    EXIT  TST                  ; AND EXIT TEST
                                     TRAP   C$EXIT
                                     .WORD  L10036-.
5852 036410                    30$:
5853 036410 032700 001000          BIT   @FATL,R0            ; DEVICE OR UNIBUS ERROR?
5854 036414 001406                    BEQ   35$                  ; NO
5855 036416                    ERRHRD 018.,ERR048,MSG011   ; YES, REPORT ERROR
                                     TRAP   C$ERHRD
                                     .WORD  18
                                     .WORD  ERR048
                                     .WORD  MSG011
5856 036426                    EXIT  TST                  ; AND EXIT TEST
                                     TRAP   C$EXIT
                                     .WORD  L10036-.
5857
5858 036432                    35$:
5859 036432 004737 027420'          JSR   PC,CHKSTR           ; SELF TEST SUCCESSFUL ?
5860 036436 103017                    BCC   40$                  ; YES
5861 036440 013704 016600'          MOV   ECODE,R4           ; NO, SET UP TO PRINT ERROR
5862 036444 006304                    ASL   R4                  ; SHIFT CODE FOR INDEX
5863 036446 062704 036560'          ADD   @STTBL,R4          ; INDEX INTO SELF TEST TABLE
5864 036452 011437 036556'          MOV   (R4),STMSG         ; LOAD INTO SELF TEST MESSAGE
5865 036456 004737 026652'          JSR   PC,CHKFTL          ; FATL BIT SET?
5866 036462                    ERRHRD 020.,ERR005,MSG004   ; REPORT SELF TEST FAILURE
                                     TRAP   C$ERHRD
                                     .WORD  20
                                     .WORD  ERR005
                                     .WORD  MSG004
5867 036472                    ESCAPE TST                 ; AND ABORT TEST
                                     TRAP   C$ESCAPE
                                     .WORD  L10036-.
5868
5869 036476 004737 030264'          40$: JSR   PC,CLR0NI        ; WRITE ONE TO CLEAR DNI
5870                                     ; ERROR?
5871 036502 103010                    BCC   50$                  ; NO
5872 036504 004737 026652'          JSR   PC,CHKFTL          ; FATL BIT SET
5873 036510                    ERRHRD 021.,ERR006,MSG003   ; YES, REPORT ERROR
                                     TRAP   C$ERHRD
                                     .WORD  21
                                     .WORD  ERR006
                                     .WORD  MSG003
5874 036520                    ESCAPE TST                 ; AND ABORT

```



5887			;LOCAL STORAGE FOR TEST 8		
5888	036556	000000	STMSG:	.WORD 0	; SELF TEST MESSAGE ADDRESS
5889			;SELF TEST MESSAGE TABLE		
5890	036560	036722'	STTBL:	.WORD SMSG00	
5891	036562	036745'		.WORD SMSG01	
5892	036564	036761'		.WORD SMSG02	
5893	036566	036775'		.WORD SMSG03	
5894	036570	037011'		.WORD SMSG04	
5895	036572	037025'		.WORD SMSG05	
5896	036574	037041'		.WORD SMSG06	
5897	036576	037055'		.WORD SMSG07	
5898	036600	037071'		.WORD SMSG10	
5899	036602	037125'		.WORD SMSG11	
5900	036604	037154'		.WORD SMSG12	
5901	036606	037170'		.WORD SMSG13	
5902	036610	037171'		.WORD SMSG14	
5903	036612	037172'		.WORD SMSG15	
5904	036614	037173'		.WORD SMSG16	
5905	036616	037174'		.WORD SMSG17	
5906	036620	037175'		.WORD SMSG20	
5907	036622	037234'		.WORD SMSG21	
5908	036624	037273'		.WORD SMSG22	
5909	036626	037322'		.WORD SMSG23	
5910	036630	037360'		.WORD SMSG24	
5911	036632	037417'		.WORD SMSG25	
5912	036634	037456'		.WORD SMSG26	
5913	036636	037523'		.WORD SMSG27	
5914	036640	037562'		.WORD SMSG30	
5915	036642	037563'		.WORD SMSG31	
5916	036644	037564'		.WORD SMSG32	
5917	036646	037565'		.WORD SMSG33	
5918	036650	037566'		.WORD SMSG34	
5919	036652	037567'		.WORD SMSG35	
5920	036654	037570'		.WORD SMSG36	
5921	036656	037571'		.WORD SMSG37	
5922	036660	037572'		.WORD SMSG40	
5923	036662	037606'		.WORD SMSG41	
5924	036664	037622'		.WORD SMSG42	
5925	036666	037636'		.WORD SMSG43	
5926	036670	037652'		.WORD SMSG44	
5927	036672	037666		.WORD SMSG45	
5928	036674	037702'		.WORD SMSG46	
5929	036676	037733'		.WORD SMSG47	
5930	036700	037764'		.WORD SMSG50	
5931	036702	040000'		.WORD SMSG51	
5932	036704	040014'		.WORD SMSG52	
5933	036706	040030'		.WORD SMSG53	
5934	036710	040044'		.WORD SMSG54	
5935	036712	040060'		.WORD SMSG55	
5936	036714	040074'		.WORD SMSG56	
5937	036716	040110'		.WORD SMSG57	
5938	036720	040124'		.WORD SMSG60	
5939					

```
5941 ;ASCII MESSAGES
5942 036722 120 101 123 MSG00: .ASCIZ /PASSED SELF TEST/<15><12>
      036725 123 105 104
      036730 040 123 105
      036733 114 106 040
      036736 124 105 123
      036741 124 015 012
      036744 000
5943 036745 125 116 104 MSG01: .ASCIZ /UNDEFINED/<15><12>
      036750 105 106 111
      036753 116 105 104
      036756 015 012 000
5944 036761 125 116 104 MSG02: .ASCIZ UNDEFINED/<15><12>
      036764 105 106 111
      036767 116 105 104
      036772 015 012 000
5945 036775 125 116 104 MSG03: .ASCIZ /UNDEFINED/<15><12>
      037000 105 106 111
      037003 116 105 104
      037006 015 012 000
5946 037011 125 116 104 MSG04: .ASCIZ /UNDEFINED/<15><12>
      037014 105 106 111
      037017 116 105 104
      037022 015 012 000
5947 037025 125 116 104 MSG05: .ASCIZ /UNDEFINED/<15><12>
      037030 105 106 111
      037033 116 105 104
      037036 015 012 000
5948 037041 125 116 104 MSG06: .ASCIZ /UNDEFINED/<15><12>
      037044 105 106 111
      037047 116 105 104
      037052 015 012 000
5949 037055 125 116 104 MSG07: .ASCIZ /UNDEFINED/<15><12>
      037060 105 106 111
      037063 116 105 104
      037066 015 012 000
5950 037071 120 110 131 MSG10: .ASCIZ /PHYSICAL ADDRESS ROM TEST/<15><12>
      037074 123 111 103
      037077 101 114 040
      037102 101 104 104
      037105 122 105 123
      037110 123 040 122
      037113 117 115 040
      037116 124 105 123
      037121 124 015 012
      037124 000
5951
5952 037125 124 111 115 MSG11: .ASCIZ /TIMER INTERRUPT TEST/<15><12>
      037130 105 122 040
      037133 111 116 124
      037136 105 122 122
      037141 125 120 124
      037144 040 124 105
      037147 123 124 015
      037152 012 000
5953 037154 125 116 104 MSG12: .ASCIZ /UNDEFINED/<15><12>
      037157 105 106 111
```

	037162	116	105	104	
	037165	015	012	000	
5954	037170	000			MSG13: .ASCIZ //
5955	037171	000			MSG14: .ASCIZ //
5956	037172	000			MSG15: .ASCIZ //
5957	037173	000			MSG16: .ASCIZ //
5958	037174	000			MSG17: .ASCIZ //
5959	037175	114	101	116	MSG20: .ASCIZ /LANCE INTERNAL LOOPBACK TEST/<15><12>
	037200	103	105	040	
	037203	111	116	124	
	037206	105	122	116	
	037211	101	114	040	
	037214	114	117	117	
	037217	120	102	101	
	037222	103	113	040	
	037225	124	105	123	
	037230	124	015	012	
	037233	000			
5960	037234	114	101	116	MSG21: .ASCIZ /LANCE IBUS PARITY ERROR TEST/<15><12>
	037237	103	105	040	
	037242	111	102	125	
	037245	123	040	120	
	037250	101	122	111	
	037253	124	131	040	
	037256	105	122	122	
	037261	117	122	040	
	037264	124	105	123	
	037267	124	015	012	
	037272	000			
5961	037273	114	101	116	MSG22: .ASCIZ /LANCE CRC LOGIC TEST/<15><12>
	037276	103	105	040	
	037301	103	122	103	
	037304	040	114	117	
	037307	107	111	103	
	037312	040	124	105	
	037315	123	124	015	
	037320	012	000		
5962	037322	114	101	116	MSG23: .ASCIZ /LANCE COLLISION DETECT TEST/<15><12>
	037325	103	105	040	
	037330	103	117	114	
	037333	114	111	123	
	037336	111	117	116	
	037341	040	104	105	
	037344	124	105	103	
	037347	124	040	124	
	037352	105	123	124	
	037355	015	012	000	
5963	037360	114	101	116	MSG24: .ASCIZ /LANCE MULTICAST ADDRESS TEST/<15><12>
	037363	103	105	040	
	037366	115	125	114	
	037371	124	111	103	
	037374	101	123	124	
	037377	040	101	104	
	037402	104	122	105	
	037405	123	123	040	
	037410	124	105	123	
	037413	124	015	012	



	037416	000			
5964	037417	114	101	116	SMSG25: .ASCIZ /LANCE BROADCAST ADDRESS TEST/<15><12>
	037422	103	105	040	
	037425	102	122	117	
	037430	101	104	103	
	037433	101	123	124	
	037436	040	101	104	
	037441	104	122	105	
	037444	123	123	040	
	037447	124	105	123	
	037452	124	015	012	
	037455	000			
5965	037456	114	101	116	SMSG26: .ASCIZ /LANCE PHYSICAL ADDRESS REJECT TEST/<15><12>
	037461	103	105	040	
	037464	120	110	131	
	037467	123	111	103	
	037472	101	114	040	
	037475	101	104	104	
	037500	122	105	123	
	037503	123	040	122	
	037506	105	112	105	
	037511	103	124	040	
	037514	124	105	123	
	037517	124	015	012	
	037522	000			
5966	037523	114	101	116	SMSG27: .ASCIZ /LANCE EXTERNAL LOOPBACK TEST/<15><12>
	037526	103	105	040	
	037531	105	130	124	
	037534	105	122	116	
	037537	101	114	040	
	037542	114	117	117	
	037545	120	102	101	
	037550	103	113	040	
	037553	124	105	123	
	037556	124	015	012	
	037561	000			
5967	037562	000			SMSG30: .ASCIZ //
5968	037563	000			SMSG31: .ASCIZ //
5969	037564	000			SMSG32: .ASCIZ //
5970	037565	000			SMSG33: .ASCIZ //
5971	037566	000			SMSG34: .ASCIZ //
5972	037567	000			SMSG35: .ASCIZ //
5973	037570	000			SMSG36: .ASCIZ //
5974	037571	000			SMSG37: .ASCIZ //
5975	037572	125	116	104	SMSG40: .ASCIZ /UNDEFINED/<15><12>
	037575	105	106	111	
	037600	116	105	104	
	037603	015	012	000	
5976	037606	125	116	104	SMSG41: .ASCIZ /UNDEFINED/<15><12>
	037611	105	106	111	
	037614	116	105	104	
	037617	015	012	000	
5977	037622	125	115	104	SMSG42: .ASCIZ /UNDEFINED/<15><12>
	037625	105	106	111	
	037630	116	105	104	
	037633	015	012	000	
5978	037636	125	116	104	SMSG43: .ASCIZ /UNDEFINED/<15><12>

	037641	105	106	111	
	037644	116	105	104	
	037647	015	012	000	
5979	037652	125	116	104	MSG44: .ASCIZ /UNDEFINED/<15><12>
	037655	105	106	111	
	037660	116	105	104	
	037663	015	012	000	
5980	037666	125	116	104	MSG45: .ASCIZ /UNDEFINED/<15><12>
	037671	105	106	111	
	037674	116	105	104	
	037677	015	012	000	
5981	037702	104	115	101	MSG46: .ASCIZ /DMA UNIBUS ACCESS TEST/<15><12>
	037705	040	125	116	
	037710	111	102	125	
	037713	123	040	101	
	037716	103	103	105	
	037721	123	123	040	
	037724	124	105	123	
	037727	124	015	012	
	037732	000			
5982	037733	104	115	101	MSG47: .ASCIZ /DMA UNIBUS ACCESS TEST/<15><12>
	037736	040	125	116	
	037741	111	102	125	
	037744	123	040	101	
	037747	103	103	105	
	037752	123	123	040	
	037755	124	105	123	
	037760	124	015	012	
	037763	000			
5983					
5984	037764	125	116	104	MSG50: .ASCIZ /UNDEFINED/<15><12>
	037767	105	106	111	
	037772	116	105	104	
	037775	015	012	000	
5985	040000	125	116	104	MSG51: .ASCIZ /UNDEFINED/<15><12>
	040003	105	106	111	
	040006	116	105	104	
	040011	015	012	000	
5986	040014	125	116	104	MSG52: .ASCIZ /UNDEFINED/<15><12>
	040017	105	106	111	
	040022	116	105	104	
	040025	015	012	000	
5987	040030	125	116	104	MSG53: .ASCIZ /UNDEFINED/<15><12>
	040033	105	106	111	
	040036	116	105	104	
	040041	015	012	000	
5988	040044	125	116	104	MSG54: .ASCIZ /UNDEFINED/<15><12>
	040047	105	106	111	
	040052	116	105	104	
	040055	015	012	000	
5989	040060	125	116	104	MSG55: .ASCIZ /UNDEFINED/<15><12>
	040063	105	106	111	
	040066	116	105	104	
	040071	015	012	000	
5990	040074	125	116	104	MSG56: .ASCIZ /UNDEFINED/<15><12>
	040077	105	106	111	
	040102	116	105	104	

	040105	015	012	000	
5991	040110	125	116	104	MSG57: .ASCIZ /UNDEFINED/<15><12>
	040113	105	106	111	
	040116	116	105	104	
	040121	015	012	000	
5992					
5993	040124	104	105	114	MSG60: .ASCIZ /DELUA IBUS LOADING TEST - CLOG/<15><12>
	040127	125	101	040	
	040132	111	102	125	
	040135	123	040	114	
	040140	117	101	104	
	040143	111	116	107	
	040146	040	124	105	
	040151	123	124	040	
	040154	055	040	103	
	040157	114	117	107	
	040162	015	012	000	
5994					.EVEN

5996  
 5997  
 5998  
 5999  
 6000  
 6001  
 6002  
 6003  
 6004  
 6005  
 6006  
 6007  
 6008  
 6009  
 6010  
 6011  
 6012  
 6013  
 6014  
 6015  
 6016  
 6017  
 6018  
 6019

.SBTTL TEST 9: PORT COMMAND TEST

```

*****
:
: THIS TEST VERIFIES THAT NO ERRORS OCCUR WHEN
: A DELUA PORT COMMAND IS ISSUED.
:
: TEST SEQUENCE:
:   1. ISSUE A DEVICE RESET
:   2. WAIT FOR DNI
:   3. WRITE A ONE TO CLEAR DNI
:   4. ISSUE A NOP PORT COMMAND
:   5. WAIT FOR DNI
:   6. WRITE ONE TO CLEAR DNI
:   7. MOVE NOP FUNCTION INTO PCBB
:   8. ISSUE A GETPCBB PORT COMMAND
:   9. WAIT FOR DNI
:  10. WRITE ONE TO CLEAR DNI
:  11. ISSUE A GETCMD PORT COMMAND
:  12. WAIT FOR DNI
:  13. WRITE ONE TO CLEAR DNI
:
*****
    
```

6020 040166  
 040166  
 6021  
 6022 040166

BGNTST

T9::

PNTMAC T09ID

040166 012704 040570'  
 040172 004737 032734'

```

MOV #T09ID,R4 ;GET POINTER TO TEST NAME MESSAGE
JSR PC,PNTID ;PRINT TEST NUMBER AND NAME
    
```

END OF MACRO EXPANSION OF 'PNTMAC'

6023 040176 004737 033134'  
 6024 040202 103034  
 6025 040204 012777 004100 140012  
 6026 040212 112777 000140 140004  
 6027 040220 004737 027736'  
 6028 040224 103010  
 6029 040226 004737 026652'  
 6030 040232  
 040232 104456  
 040234 000026  
 040236 023030'  
 040240 021736'

```

JSR PC,TINIT ; IS A DEVICE RESET NEEDED?
BCC 25$ ; NO
MOV #DNI+INTE,@PCSR0 ; PRECONDITION INTR EN.
MOVB #INTE!RSET,@PCSR0 ; RESET DELUA
JSR PC,CKDNI ; DNI ?
BCC 20$ ; YES
JSR PC,CHKFTL ; FATL BIT SET?
ERRHRD 022.,ERR006,MSG003 ; REPORT ERROR
    
```

TRAP C\$ERHRD  
 .WORD 22  
 .WORD ERR006  
 .WORD MSG003

6031 040242  
 040242 104410  
 040244 000350

ESCAPE TST ; AND ABORT TEST

TRAP C\$ESCAPE  
 .WORD L10037 .

6032  
 6033 040246 004737 030264'  
 6034  
 6035 040252 103010  
 6036 040254 004737 026652'  
 6037 040260  
 040260 104456  
 040262 000027

```

: 20$: JSR PC,CLRDN1 ; WRITE ONE TO CLEAR DNI
: ; ERROR
BCC 25$ ; NO
JSR PC,CHKFTL ; FATL BIT SET
ERRHRD 023.,ERR006,MSG003 ; YES, REPORT ERROR
    
```

TRAP C\$ERHRD  
 .WORD 23

	040264	023030'							.WORD	ERR006
	040266	021736'							.WORD	MSG003
6038	040270		ESCAPE	TST			; AND ABORT		TRAP	C\$ESCAPE
	040270	104410							.WORD	L10037-
	040272	000322							.WORD	
6039										
6040	040274	012777	004100	137722	25\$:	MOV	#DNI+INTE,@PCSR0	; PRE-CONDITION INTR EN.		
6041	040302	112777	000106	137714		MOVB	#INTE!PNOP,@PCSR0	; ISSUE A NOP PORT COMMAND		
6042	040310	004737	026550'			JSR	PC,CHKDNI	; DNI ?		
6043	040314	103010				BCC	30\$	; YES		
6044	040316	004737	026652'			JSR	PC,CHKFTL	; FATL BIT SET?		
6045	040322					ERRHRD	024.,ERR008,MSG003	; NO, REPORT ERROR		
	040322	104456							TRAP	C\$ERHRD
	040324	000030							.WORD	24
	040326	023166'							.WORD	ERR008
	040330	021736'							.WORD	MSG003
6046	040332		ESCAPE	TST			; AND ABORT TEST		TRAP	C\$ESCAPE
	040332	104410							.WORD	L10037-
	040334	000260								
6047										
6048	040336	004737	030264'		30\$:	JSR	PC,CLRDN1	; WRITE ONE TO CLEAR DNI		
6049								; ERROR		
6050	040342	103010				BCC	40\$	; NO		
6051	040344	004737	026652'			JSR	PC,CHKFTL	; FATL BIT SET?		
6052	040350					ERRHRD	025.,ERR006,MSG003	; YES, REPORT ERROR		
	040350	104456							TRAP	C\$ERHRD
	040352	000031							.WORD	25
	040354	023030'							.WORD	ERR006
	040356	021736'							.WORD	MSG003
6053	040360		ESCAPE	TST			; AND ABORT		TRAP	C\$ESCAPE
	040360	104410							.WORD	L10037 .
	040362	000232								
6054										
6055	040364				40\$:	MOV	#NOPF,R5	; POINT TO DEFAULT NOP FUNCTION		
6056	040364	012705	012440'			JSR	PC,LDPCCB	; LOAD FUNCTION INTO PCBB		
6057	040370	004737	032002'			JSR	PC,LDPCSR	; ADDRESS OF PCBB -> PCSR2!3		
6058	040374	004737	032032'			MOV	#DNI+INTE,@PCSR0	; PRECONDITION INTR EN.		
6059	040400	012777	004100	137616		MOVB	#INTE!GETPCB,@PCSR0	; ISSUE A GETPCBB PORT COMMAND		
6060	040406	112777	000101	137610		JSR	PC,CHKDNI	; DNI?		
6061	040414	004737	026550'			BCC	50\$	; YES		
6062	040420	103010				JSR	PC,CHKFTL	; FATL BIT SET		
6063	040422	004737	026652'			ERRHRD	026.,ERR009,MSG003	; NO, REPORT ERROR		
6064	040426								TRAP	C\$ERHRD
	040426	104456							.WORD	26
	040430	000032							.WORD	ERR009
	040432	023245'							.WORD	MSG003
	040434	021736'								
6065	040436		ESCAPE	TST			; AND ABORT TEST		TRAP	C\$ESCAPE
	040436	104410							.WORD	L10037-
	040440	000154								
6066										
6067	040442	004737	030264'		50\$:	JSR	PC,CLRDN1	; WRITE ONE TO CLEAR DNI		
6068								; ERROR ?		
6069	040446	103010				BCC	60\$	; NO		
6070	040450	004737	026652'			JSR	PC,CHKFTL	; FATL BIT SET?		
6071	040454					ERRHRD	027.,ERR006,MSG003	; YES, REPORT ERROR		
	040454	104456							TRAP	C\$ERHRD

```

040456 000033 .WORD 27
040460 023030' .WORD ERR006
040462 021736' .WORD MSG003
6072 040464 ESCAPE TST ; AND ABORT TEST
040464 104410 TRAP C$ESCAPE
040466 000126 .WORD L10037
6073
6074 040470 012777 004100 137526 ; 60$: MOV #DNI+INTE,@PCSR0 ; PRECONDITION INTR EN.
6075 040476 112777 000102 137520 MOVB #INTE!GETCMD,@PCSR0 ; ISSUE A GETCMD PORT COMMAND
6076 040504 004737 026550' JSR PC,CHKDNI ; DNI ?
6077 040510 103010 BCC 70$ ; YES
6078 040512 004737 026652' JSR PC,CHKFTL ; FATL BIT SET?
6079 040516 ERRHRD 030.,ERR010,MSG003 ; NO, REPORT ERROR
040516 104456 TRAP C$ERHRD
040520 000036 .WORD 30
040522 023331' .WORD ERR010
040524 021736' .WORD MSG003
6080 040526 ESCAPE TST ; AND ABORT TEST
040526 104410 TRAP C$ESCAPE
040530 000064 .WORD L10037-.
6081
6082 040532 004737 030264' ; 70$: JSR PC,CLRDN1 ; WRITE ONE TO CLEAR DNI
6083 BCC 80$ ; ERROR ?
6084 040536 103010 JSR PC,CHKFTL ; NO
6085 040540 004737 026652' ERRHRD 031.,ERR006,MSG003 ; FATL BIT SET?
6086 040544 104456 ; YES, REPORT ERROR
040544 104456 TRAP C$ERHRD
040546 000037 .WORD 31
040550 023030' .WORD ERR006
040552 021736' .WORD MSG003
6087 040554 ESCAPE TST ; AND ABORT TEST
040554 104410 TRAP C$ESCAPE
040556 000036 .WORD L10037 .
6088 040560 ; 80$: JSR PC,CLINTR ; INSURE DELUA INTR BITS SET
6089 040560 004737 030346'
6090 EXIT TST
6091 040564 TRAP C$EXIT
040564 104432 .WORD L10037 .
040566 000026
6092 ;LOCAL TEST MESSAGE
6093
6094
6095 040570 104 105 114 T09ID: .ASCIZ 'DELUA PORT COMMAND '
040573 125 101 040
040576 120 117 122
040601 124 040 103
040604 117 115 115
040607 101 116 104
040612 040 000
6096 .EVEN
6097
6098 040614 ENDTST
040614 L10037: TRAP C$ETST
040614 104401
    
```

6100  
 6101  
 6102  
 6103  
 6104  
 6105  
 6106  
 6107  
 6108  
 6109  
 6110  
 6111  
 6112

.SBTTL TEST 10: INTERRUPT LOGIC TEST

```

;*****
;
; THIS TEST VERIFIES THAT A DELUA INTERRUPT CAN BE GENERATED.
;
; TEST SEQUENCE:
;   1. SET UP THE INTERRUPT VECTOR
;   2. ISSUE A GET PCBB PORT COMMAND
;   3. WAIT FOR A DNI INTERRUPT
;   4. WRITE ONE TO CLEAR DNI
;*****
    
```

6114 040616  
 040616  
 6115  
 6116 040616

BGNTST

T10::

PNTMAC T10ID

040616 012704 041136'  
 040622 004737 032734'

```

MOV   #T10ID,R4      ;GET POINTER TO TEST NAME MESSAGE
JSR   PC,PNTID       ;PRINT TEST NUMBER AND NAME
    
```

; END OF MACRO EXPANSION OF 'PNTMAC'

6117 040626 004737 033434'  
 6118 040632 103034  
 6119 040634 012777 004100 137362  
 6120 040642 112777 000140 137354  
 6121 040650 004737 027736'  
 6122 040654 103010  
 6123 040656

```

JSR   PC,TINIT      ; IS A DEVICE RESET NEEDED?
BCC   25$           ; NO
MOV   #DNI+INTE,@PCSR0 ; PRECONDITION INTR EN.
MOVB  #INTE!RSET,@PCSR0 ; YES, RESET DELUA
JSR   PC,CKDNI      ; DNI ?
BCC   20$           ; YES
FTL
    
```

040656 004737 026652'

```

JSR   PC,CHKFTL     ; 'FATL' BIT SET?
    
```

6124 040662  
 040662 104456  
 040664 000040  
 040666 023030'  
 040670 021736'  
 6125 040672  
 040672 104410  
 040674 000272

```

ERRHRD 032.,ERR006,MSG003 ; NO, REPORT ERROR
    
```

```

TRAP   C$ERHRD
.WORD  32
.WORD  ERR006
.WORD  MSG003
    
```

```

ESCAPE TST          ; AND ABORT TEST
    
```

```

TRAP   C$ESCAPE
.WORD  L10040-
    
```

6126  
 6127 040676 004737 030264'  
 6128  
 6129 040702 103010  
 6130 040704

; 20\$:

```

JSR   PC,CLRDNI     ; WRITE ONE TO CLEAR DNI
; ERROR
; NO
    
```

040704 004737 026652'

```

JSR   PC,CHKFTL     ; 'FATL' BIT SET?
    
```

6131 040710  
 040710 104456  
 040712 000041  
 040714 023030'  
 040716 021736'

```

ERRHRD 033.,ERR006,MSG003 ; YES, REPORT ERROR
    
```

```

TRAP   C$ERHRD
.WORD  33
.WORD  ERR006
.WORD  MSG003
    
```

6132 040720  
 040720 104410

```

ESCAPE TST          ; AND ABORT
    
```

```

TRAP   C$ESCAPE
    
```

```

        040722 000244                                .WORD L10040 .
6133
6134 ;SET UP INTERRUPT VECTOR
6135
6136 040724 000244                                25$: SETVEC INTVEC, #ISRDN1, UNAPRI
        040724 013746 000242'                                MOV UNAPRI, -(SP)
        040730 012746 034374'                                MOV #ISRDN1, -(SP)
        040734 013746 000240'                                MOV INTVEC, -(SP)
        040740 012746 000003                                MOV #3, -(SP)
        040744 104437                                TRAP C#SVEC
        040746 062706 000010                                ADD #10, SP
6137 040752                                SETPRI #PRI04 ; SET CPU PRIORITY = 4
        040752 012700 000200                                MOV #PRI04, R0
        040756 104441                                TRAP C#SPRI
6138
6139 ;ISSUE GET PCBB PORT COMMAND WITH INTERRUPTS ENABLED
6140
6141 040760                                40$:
6142 040760 005037 016610'                                CLR DNIFLG ; INSURE DNI BIT SET FLAG IS CLEAR
6143 040764 012705 012440'                                MOV #NOPF, R5 ; POINT TO DEFAULT NOP FUNCTION
6144 040770 004737 032002'                                JSR PC, LDPCBB ; LOAD FUNCTION INTO PCBB
6145 040774 004737 032032'                                JSR PC, LDPCSR ; ADDRESS OF PCBB -> PCSR2!3
6146 041000 012777 004100 137216                                MOV #DNI!INTE, @PCSR0 ; PRECONDITION INTR EN.
6147 041006 112777 000101 137210                                MOV #INTE!GETPCB, @PCSR0 ; ISSUE A GETPCBB PORT COMMAND
6148
6149 ;WAIT FOR DNI INTERRUPT
6150
6151 041014 012701 005000                                50$: MOV #5000, R1 ; INIT WAIT COUNT
6152 041020
6153 041020 005737 016610'                                TST DNIFLG ; DID DNI INTERRUPT OCCUR?
6154 041024 001020                                BNE 70$ ; YES, CONTINUE TEST
6155 041026 005301                                DEC R1 ; REDUCE DELAY
6156 041030 001373                                BNE 50$ ; NOT YET
6157 041032
        041032 004737 026652'                                JSR PC, CHKFTL ; 'FATL' BIT SET?
        ERRHRD 034., ERRO07 ; YES, REPORT ERROR
6158 041036                                TRAP C#ERHRD
        041036 104456                                .WORD 34
        041040 000042                                .WORD ERRO07
        041042 023076'                                .WORD 0
        041044 000000
6159 041046                                CLRVEC INTVEC ; DEALLOCATE VECTOR
        041046 013700 000240'                                MOV INTVEC, R0
        041052 104436                                TRAP C#CVEC
6160 041054                                SETPRI #PRI07 ; RESTORE CPU PRIORITY TO 7
        041054 012700 000340                                MOV #PRI07, R0
        041060 104441                                TRAP C#SPRI
6161 041062                                ESCAPE TST ; AND ABORT TEST
        041062 104410                                TRAP C#ESCAPE
        041064 000102                                .WORD L10040-.
6162
6163 ;WRITE ONE TO CLEAR DNI
6164
6165 041066 004737 033402'                                70$: JSR PC, TIMOFF ; TURN OFF THE TIMER
6166 041072                                SETPRI #PRI07 ; RESTORE CPU PRIORITY TO 7
        041072 012700 000340                                MOV #PRI07, R0
    
```



```

041076 104441
6167 041100 004737 030264' JSR PC,CLRDNI ; WRITE ONE TO CLEAR DNI TRAP C$SPRI
6168 ; ERROR?
6169 041104 103010 BCC 80$ ; NO
6170 041106 FTL

041106 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

6171 041112 ERRHRD 035.,ERR006,MSG003 ; YES, REPORT ERROR TRAP C$ERHRD
041112 104456 .WORD 35
041114 000043 .WORD ERR006
041116 023030' .WORD MSG003
041120 021736'

6172 041122 ESCAPE TST ; AND ABORT TRAP C$ESCAPE
041122 104410 .WORD L10040-.
041124 000042

6173 041126 80$: JSR PC,CLINTR ; INSURE DELUA INTR BITS CLEAR
6174 041126 004737 030346'
6175
6176 041132 EXIT TST TRAP C$EXIT
041132 104432 .WORD L10040-.
041134 000032

6177
6178 ;LOCAL TEST MESSAGE
6179
6180 041136 104 105 114 T10ID:.ASCIZ 'DELUA INTERRUPT LOGIC '
041141 125 101 040
041144 111 116 124
041147 105 122 122
041152 125 120 124
041155 040 114 117
041160 107 111 103
041163 040 000

6181 .EVEN
6182
6183 041166 ENDTST
041166 L10040: TRAP C$ETST
041166 104401
  
```

6185  
 6186  
 6187  
 6188  
 6189  
 6190  
 6191  
 6192  
 6193  
 6194  
 6195  
 6196  
 6197  
 6198  
 6199  
 6200  
 6201  
 6202  
 6203

.SBTTL TEST 11: READ INTERNAL ROM TEST

```

*****
:
: THIS TEST READS AND VERIFIES THE INTERNAL ROM.
: THE DUMP INTERNAL MEMORY FUNCTION IS USED TO READ THE ROM.
: A CRC IS GENERATED FROM THE ROM DATA READ.
: A CRC VALUE OF ZERO SHOULD BE GENERATED FROM THE ROM
: DATA READ WHICH INCLUDES THE STORED ROM CRC VALUE.
:
: TEST SEQUENCE:
:   1. CLEAR RBUF
:   2. READ 1K OF ROM INTO RBUF
:   3. CALCULATE CRC ON RBUF
:   4. REPEAT STEPS 1-3 FOR EACH 1K BLOCK OF ROM (8 TIMES)
:   5. VERIFY CRC GENERATED = 0
:
*****
  
```

6204 041170  
 041170  
 6205  
 6206 041170

BGNTST

T11::

PNTMAC T11ID

041170 012704 041664'  
 041174 004737 032734'

MOV #T11ID,R4  
 JSR PC,PNTID

;GET POINTER TO TEST NAME MESSAGE  
 ;PRINT TEST NUMBER AND NAME

; END OF MACRO EXPANSION OF 'PNTMAC'

6207 041200 004737 033434'  
 6208 041204 103034  
 6209 041206 012777 004100 137010  
 6210 041214 112777 000140 137002  
 6211 041222 004737 027736'  
 6212 041226 103010  
 6213 041230

JSR PC,TINIT  
 BCC 30\$  
 MOV #DNI!INTE,@PCSR0  
 MOVB #INTE!RSET,@PCSR0  
 JSR PC,CKDNI  
 BCC 20\$  
 FTL

; IS A DEVICE RESET NEEDED?  
 ; NO  
 ; PRECONDITION INTR EN.  
 ; YES, RESET DELUA  
 ; DNI ?  
 ; YES

041230 004737 026652'

JSR PC,CHKFTL

; 'FATL' BIT SET?

6214 041234  
 041234 104456  
 041236 000044  
 041240 023030'  
 041242 021736'

ERRHRD 036.,ERR006,MSG003

; NO, REPORT ERROR

TRAP C\$ERHRD  
 .WORD 36  
 .WORD ERR006  
 .WORD MSG003

6215 041244  
 041244 104410  
 041246 000450

ESCAPE TST

; AND ABORT TEST

TRAP C\$ESCAPE  
 .WORD L10041-

6216  
 6217 041250 004737 030264'  
 6218  
 6219 041254 103010  
 6220 041256

; 20\$: JSR PC,CLR DNI

; WRITE ONE TO CLEAR DNI  
 ; ERROR ?  
 ; NO

041256 004737 026652'

JSR PC,CHKFTL

; 'FATL' BIT SET?

6221 041262  
 041262 104456

ERRHRD 037.,ERR006,MSG003

; YES, REPORT ERROR

TRAP C\$ERHRD

```

        041264 000045                .WORD 37
        041266 023030'              .WORD ERR006
        041270 021736'              .WORD MSG003
6222 041272                ESCAPE TST                ; AND ABORT TEST
        041272 104410                TRAP C$ESCAPE
        041274 000422                .WORD L10041-.
6223
6224 041276 004737 032032'          ; 30$: JSR PC,LDPCSR                ; ADDRESS OF PCBB > PCSR2!3
6225 041302 012777 004100 136714   ; MOV #DNI!INTE,@PCSR0          ; PRECONDITION INTR EN.
6226 041310 112777 000101 136706   ; MOVB #INTE!GETPCB,@PCSR0     ; ISSUE GET_PCBB PORT COMMAND
6227 041316 004737 026550'          ; JSR PC,CHKDNI                 ; DNI ?
6228 041322 103010                ; BCC 40$                       ; YES
6229 041324                FTL
        041324 004737 026652'          ; JSR PC,CHKFTL                 ; 'FATL' BIT SET?
6230 041330                ERRHRD 040.,ERR009,MSG003          ; NO, REPORT ERROR
        041330 104456                TRAP C$ERHRD
        041332 000050                .WORD 40
        041334 023245'              .WORD ERR009
        041336 021736'              .WORD MSG003
6231 041340                ESCAPE TST                ; AND ABORT TEST
        041340 104410                TRAP C$ESCAPE
        041342 000354                .WORD L10041 .
6232
6233 041344 004737 030264'          ; 40$: JSR PC,CLRDN1             ; WRITE ONE TO CLEAR DNI
6234                                ; BCC 50$                       ; ERROR ?
6235 041350 103010                ; FTL                            ; NO
6236 041352                JSR PC,CHKFTL                 ; 'FATL' BIT SET?
        041352 004737 026652'          ; ERRHRD 041.,ERR006,MSG003    ; YES, REPORT ERROR
6237 041356                TRAP C$ERHRD
        041356 104456                .WORD 41
        041360 000051                .WORD ERR006
        041362 023030'              .WORD MSG003
        041364 021736'
6238 041366                ESCAPE TST                ; AND ABORT TEST
        041366 104410                TRAP C$ESCAPE
        041370 000326                .WORD L10041 .
6239
6240 041372 012705 012664'          ; 50$: MOV #DMPMEM,R5           ; DEFAULT DUMP INTERNAL MEMORY
6241 041376 004737 032002'          ; JSR PC,LDPCBB                ; LOAD FUNCTION -> PCBB
6242 041402 012705 015622'          ; MOV #UDB10A,R5              ; DEFAULT UDBB
6243 041406 012700 000005          ; MOV #5,R0                    ; FOUR WORDS
6244 041412 004737 032260'          ; JSR PC,LDUDBB                ; LOAD INTO UDBB
6245
6246
6247 041416 012737 002000 016560'   ; MOV #2000,BYTCNT             ; 1K BYTES FOR SUBROUTINE 'ROMCRC'
6248 041424 012702 015634'          ; MOV #MEM10A,R2               ; R2 POINTS TO ROM ADDRESS TABLE
6249 041430 012700 016574'          ; MOV #XCRC,R0                 ; POINT TO CRC STORAGE
6250 041434 012720 177777          ; MOV #-1,(R0)+                ; SET INITIAL
6251 041440 012710 177777          ; MOV #-1,(R0)                 ; CRC
6252 041444 012701 000017          ; MOV #15.,R1                  ; PERFORM 15 ROM DUMPS
6253                                ; AND CRC CALCULATIONS
6254
6255 041450 004737 030236'          ; 60$: JSR PC,CLRCV            ; CLEAR RBUF
    
```

```

6256
6257 041454 012237 000316'      ;
6258 041460 012777 004100 136536  MOV      (R2)+,UDBB+6      ; LOAD ROM ADDRESS > UDBB+6
6259 041466 112777 000102 136530  MOV      #DNI!INTE,@PCSR0 ; PRECONDITION INTR EN.
6260 041474 004737 026550'      MOVVB   #INTE!GETCMD,@PCSR0 ; ISSUE GET COMMAND PORT COMMAND
6261 041500 103010                JSR      PC,CHKDNI        ; DNI ?
6262 041502                BCC     70$              ; YES
                          FTL
                          JSR      PC,CHKFTL        ; 'FATL' BIT SET?
                          ERRHRD   042.,ERR010,MSG003 ; NO, REPORT ERROR
6263 041506                TRAP    C$ERHRD
041506 104456                .WORD  42
041510 000052                .WORD  ERR010
041512 023331'              .WORD  MSG003
041514 021736'
6264 041516                ESCAPE  TST              ; AND ABORT TEST
041516 104410                TRAP    C$ESCAPE
041520 000176                .WORD  L10041-.
6265
6266 041522 004737 030264'      ;70$: JSR      PC,CLRDN1        ; WRITE ONE TO CLEAR DNI
6267                                ; ERROR ?
6268 041526 103010                BCC     80$              ; NO
6269 041530                FTL
                          JSR      PC,CHKFTL        ; 'FATL' BIT SET?
                          ERRHRD   043.,ERR006,MSG003 ; YES, REPORT ERROR
6270 041534                TRAP    C$ERHRD
041534 104456                .WORD  43
041536 000053                .WORD  ERR006
041540 023030'              .WORD  MSG003
041542 021736'
6271 041544                ESCAPE  TST              ; AND ABORT TEST
041544 104410                TRAP    C$ESCAPE
041546 000150                .WORD  L10041-.
6272
6273 041550                ;80$: JSR      PC,ROMCRC        ; CALCULATE CRC ON 1K RBUF
6274 041550 004737 032462'      DEC     R1                ; REDUCE 1K BLOCK COUNT
6275 041554 005301                CMP     R1,#1             ; NEXT BLOCK LAST ONE?
6276 041556 020127 000001      BGE     60$              ; NO
6277 041562 002332                TST    R1                ; ALL DONE?
6278 041564 005701                BMI     85$              ; YES
6279 041566 100404                MOV     #1774,BYTCNT     ; NO, BUT,DON'T INCLUDE CRC
6280 041570 012737 001774 016560' BR      60$              ; IN LAST 1K CRC CALCULATION
6281 041576 000724
6282                                ;
6283                                ;VERIFY CRC
6284
6285 041600                ;85$: COM     R3                ; COMPLIMENT
6286 041600 005103                COM     R4                ; CRC
6287 041602 005104                MOV     #XCRC,R0         ; BASE ADDRESS OF CALCULATED CRC
6288 041604 012700 016574'      MOV     R4,(R0)+         ; SAVE
6289 041610 010420                MOV     R3,(R0)         ; CRC
6290 041612 010310                MOV     #XCRC,R0         ; RESET POINTER
6291 041614 012700 016574'
6292
6293 041620 012701 010434'      MOV     #RBUF+1774,R1    ; POINT TO ROM CRC
6294 041624 022021                CMP     (R0)+,(R1)+     ; 1ST 2 BYTES CHECK?

```

```

6295 041626 001002          BNE      90$          ; NO, GO REPORT ERROR
6296 041630 021011          CMP      (R0),(R1)     ; 2ND 2 BYTES COMPARE?
6297 041632 001410          BEQ      95$          ; YES, GO EXIT TEST
6298 041634
6299 041634          90$:      FTL

          041634 004737 026652'      JSR      PC,CHKFTL     ; 'FATL' BIT SET?
6300 041640          ERRHRD 044.,ERR024     ; NO, ROM CRC ERROR, REPORT ERROR
          041640 104456          TRAP     C$ERHRD
          041642 000054          .WORD   44
          041644 024444'          .WORD   ERR024
          041646 000000          .WORD   0
6301 041650          ESCAPE TST          ; AND ABORT TEST
          041650 104410          TRAP     C$ESCAPE
          041652 000044          .WORD   L10041 .
6302 041654          95$:      JSR      PC,CLINTR     ; INSURE DEUNA INTR BITS CLEAR
6303 041654 004737 030346'      EXIT     TST
6304
6305 041660          EXIT     TST          TRAP     C$EXIT
          041660 104432          .WORD   L10041-.
          041662 000034
6306
6307          ;LOCAL TEST MESSAGE
6308
6309 041664          104      105      114      T11ID:.ASCIZ 'DELUA READ INTERNAL ROM '
          041667          125      101      040
          041672          122      105      101
          041675          104      040      111
          041700          116      124      105
          041703          122      116      101
          041706          114      040      122
          041711          117      115      040
          041714          000
6310          .EVEN
6311
6312 041716          ENDTST
          041716
          041716 104401          L10041: TRAP     C$ETST
  
```

6314  
 6315  
 6316  
 6317  
 6318  
 6319  
 6320  
 6321  
 6322  
 6323  
 6324  
 6325  
 6326  
 6327  
 6328  
 6329  
 6330  
 6331  
 6332  
 6333  
 6334  
 6335  
 6336  
 6337  
 6338  
 6339  
 6340  
 6341  
 6342  
 6343 041720  
 041720  
 6344  
 6345 041720  
 041720 012704 043114'  
 041724 004737 032734'  
 6346 041730 004737 033434'  
 6347 041734 103034  
 6348 041736 012777 004100 136260  
 6349 041744 112777 000140 136252  
 6350 041752 004737 027736'  
 6351 041756 103010  
 6352 041760  
 041760 004737 026652'  
 6353 041764  
 041764 104456  
 041766 000055  
 041770 023030'  
 041772 021736'  
 6354 041774  
 041774 104410  
 041776 001160

.SBTTL TEST 12: READ/WRITE INTERNAL MEMORY TEST

```

*****
:
: THIS TEST READS AND WRITES THE INTERNAL RAM MEMORY.
: THE DUMP/LOAD INTERNAL MEMORY FUNCTIONS ARE USED TO
: READ/WRITE THE ENTIRE INTERNAL RAM ABOVE THAT USED
: FOR THE LOADED PROCESS.
:
:         LOWEST ADDRESS: 08400(16)
:         HIGHEST ADDRESS: 1F400(16)
:
: TEST SEQUENCE:
:
:     1. WRITE MODE REGISTER = INTERNAL LOOPBACK MODE
:        TO REMOVE MEMORY FROM THE WIRE
:     2. LOAD TBUF WITH DATA = ADDRESS
:     3. LOAD 1K OF INTERNAL MEMORY WITH TBUF
:     4. REPEAT STEPS 1 AND 2 FOR
:        EACH 1K BLOCK OF MEMORY (    TIMES)
:     5. RESETUP TBUF FOR DATA COMPARE
:     6. CLEAR RBUF
:     7. DUMP INTERNAL MEMORY -> RBUF
:     8. COMPARE RBUF WITH TBUF
:     9. REPEAT STEPS 4,5,6 AND 7 FOR EACH 1K BLOCK
:    10. REPEAT STEPS 1 THRU 8 WITH COMPLIMENT DATA
:
*****
    
```

BGNTST

T12::

PNTMAC T12ID

```

MOV    #T12ID,R4      ;GET POINTER TO TEST NAME MESSAGE
JSR    PC,PNTID       ;PRINT TEST NUMBER AND NAME
    
```

END OF MACRO EXPANSION OF 'PNTMAC'

```

JSR    PC,TINIT       ; IS A DEVICE RESET NEEDED?
BCC    30$            ; NO
MOV    #DNI!INTE,@PCSR0 ; PRECONDITION INTR EN.
MOVB   #INTE!RSET,@PCSR0 ; YES, RESET DELUA
JSR    PC,CKDNI       ; DNI ?
BCC    20$            ; YES
FTL
    
```

```

JSR    PC,CHKFTL      ; 'FATL' BIT SET?
    
```

```

ERRHRD 045.,ERR006,MSG003 ; NO, REPORT ERROR
    
```

```

TRAP   C$ERHRD
.WORD  45
.WORD  ERR006
.WORD  MSG003
    
```

```

ESCAPE TST           ; AND ABORT TEST
    
```

```

TRAP   C$ESCAPE
.WORD  L10042-.
    
```

```

6355
6356 042000 004737 030264'      ; 20$: JSR    PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
6357                                ; ERROR ?
6358 042004 103010                BCC    30$                ; NO
6359 042006                                FTL
                                JSR    PC,CHKFTL      ; 'FATL' BIT SET?
                                ERRHRD 046.,ERR006,MSG003 ; YES, REPORT ERROR
6360 042012                                TRAP  C$ERHRD
                                042012 104456                .WORD 46
                                042014 000056                .WORD ERR006
                                042016 023030'                .WORD MSG003
                                042020 021736'
6361 042022                                ESCAPE TST                ; AND ABORT TEST
                                042022 104410                TRAP  C$ESCAPE
                                042024 001132                .WORD L10042-.
6362
6363 042026 004737 032032'      ; 30$: JSR    PC,LDPCSR      ; ADDRESS OF PCBB -> PCSR2:3
6364 042032 012777 004100 136164 MOV    #DNI!INTE,@PCSR0 ; PRECONDITION INTR EN.
6365 042040 112777 000101 136156 MOVB   #INTE!GETPCB,@PCSR0 ; ISSUE GET_PCBB PORT COMMAND
6366 042046 004737 026550'      JSR    PC,CHKDNI
6367 042052 103010                BCC    40$                ; YES
6368 042054                                FTL
                                JSR    PC,CHKFTL      ; 'FATL' BIT SET?
                                ERRHRD 047.,ERR009,MSG003 ; NO, REPORT ERROR
6369 042060                                TRAP  C$ERHRD
                                042060 104456                .WORD 47
                                042062 000057                .WORD ERR009
                                042064 023245'                .WORD MSG003
                                042066 021736'
6370 042070                                ESCAPE TST                ; AND ABORT TEST
                                042070 104410                TRAP  C$ESCAPE
                                042072 001064                .WORD L10042-.
6371
6372 042074 004737 030264'      ; 40$: JSR    PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
6373                                ; ERROR ?
6374 042100 103010                BCC    45$                ; NO
6375 042102                                FTL
                                JSR    PC,CHKFTL      ; 'FATL' BIT SET?
                                ERRHRD 050.,ERR006,MSG003 ; YES, REPORT ERROR
6376 042106                                TRAP  C$ERHRD
                                042106 104456                .WORD 50
                                042110 000062                .WORD ERR006
                                042112 023030'                .WORD MSG003
                                042114 021736'
6377 042116                                ESCAPE TST                ; AND ABORT TEST
                                042116 104410                TRAP  C$ESCAPE
                                042120 001036                .WORD L10042-.
6378
6379                                ; ISSUE A PORT HALT TO INHIBIT NI ACTIVITY
6380                                ;
6381 042122                                ; 45$:
6382
6383 042122 012777 004100 136074 MOV    #DNI!INTE,@PCSR0 ; PRECONDITION INTR EN.
6384 042130 112777 000116 136066 MOVB   #INTE!HALT,@PCSR0 ; PORT HALT
    
```

```

6385 042136 004737 026550'      JSR    PC,CHKDNI      ; DNI ?
6386 042142 103010              BCC    47$           ; YES
6387 042144                      FTL

      042144 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?
6388 042150                      ERRHRD 051.,ERR038,MSG003 ; NO, REPORT ERROR
      042150 104456              TRAP   C$ERHRD
      042152 000063              .WORD 51
      042154 025522'            .WORD ERR038
      042156 021736'            .WORD MSG003
6389 042160                      ESCAPE TST           ; AND ABORT TEST
      042160 104410              TRAP   C$ESCAPE
      042162 000774              .WORD L10042-.
6390                      ;
6391 042164 004737 030264'      47$: JSR    PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
6392                      ; ERROR ?
6393 042170 103010              BCC    50$           ; NO
6394 042172                      FTL

      042172 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?
6395 042176                      ERRHRD 052.,ERR006,MSG003 ; YES, REPORT ERROR
      042176 104456              TRAP   C$ERHRD
      042200 000064              .WORD 52
      042202 023030'            .WORD ERR006
      042204 021736'            .WORD MSG003
6396 042206                      ESCAPE TST           ; AND ABORT TEST
      042206 104410              TRAP   C$ESCAPE
      042210 000746              .WORD L10042-.
6397                      ;
6398                      ;WRITE RAM MEMORY WITH DATA = ADDRESS BY 1K BLOCKS
6399 042212                      50$:
6400 042212 005037 016606'      CLR    EAFLAG        ; CLEAR EXT ADDR BITS FLAG
6401 042216 012703 015712'      MOV    #MEM13A,R3    ; R3 POINTS TO LINK MEM ADDRESS TABLE
6402 042222 012701 000065      MOV    #53.,R1       ; DO LOOP
6403                      ;
6404                      ;WRITE TBUF WITH DATA = ADDRESS
6405 042226 010305                      60$: MOV    R3,R5        ; R5 POINTS TO ADDRESS
6406 042230 004737 031446'      JSR    PC,LDBUFC     ; LOAD TBUF WITH ADDRESS DATA PATTERN
6407                      ;
6408                      ;LOAD INTERNAL RAM MEMORY
6409 042234 012705 012674'      MOV    #LDMEM,R5    ; DEFAULT LOAD INTERNAL MEMORY
6410 042240 004737 033256'      JSR    PC,SRWRAM    ; LOAD PCBB AND UDBB
6411                      ;
6412 042244 012777 004100 135752 65$: MOV    #DNI!INTE,@PCSR0 ; PRECONDITION INTR EN.
6413 042252 112777 000102 135744 MOVB   #INTE!GETCMD,@PCSR0 ; ISSUE GET COMMAND PORT COMMAND
6414 042260 004737 026550'      JSR    PC,CHKDNI      ; DNI ?
6415 042264 103010              BCC    70$           ; YES
6416 042266                      FTL

      042266 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?
6417 042272                      ERRHRD 053.,ERR010,MSG003 ; NO, REPORT ERROR
      042272 104456              TRAP   C$ERHRD
      042274 000065              .WORD 53
      042276 023331'            .WORD ERR010
    
```



```

        042300 021736'
6418 042302          ESCAPE TST          ; AND ABORT TEST          .WORD  MSG003
        042302 104410
        042304 000652
        70$: JSR      PC,CLRDNI          ; WRITE ONE TO CLEAR DNI
6419 042306 004737 030264'          ; ERROR ?
6421          BCC      80$              ; NO
6422 042312 103010          FTL
6423 042314          JSR      PC,CHKFTL          ; 'FATL' BIT SET?
        042314 004737 026652'          ERRHRD 054.,ERR006,MSG003          ; YES, REPORT ERROR
6424 042320          TRAP      C$ERHRD
        042320 104456          .WORD  54
        042322 000066          .WORD  ERR006
        042324 023030'          .WORD  MSG003
        042326 021736'
6425 042330          ESCAPE TST          ; AND ABORT TEST          TRAP      C$ESCAPE
        042330 104410          .WORD  L10042-.
        042332 000624
        80$: TST      (R3)+          ; BUMP TABLE POINTER
6426 042334          DEC      R1          ; DONE 16 WRITES ?
6427 042334 005725          BNE     60$          ; NO
6428 042336 005301
6429 042340 001332
6430          ;
6431          ;READ INTERNAL RAM MEMORY BY 1K BLOCKS AND COMPARE DATA
6432          CLR      EAFLAG          ; CLEAR EXT ADDR BITS FLAG
6433 042342 005037 016606'          MOV     #MEM13A,R3          ; R3 POINTS TO LINK MEM ADDRESS TABLE
6434 042346 012703 015712'          MOV     #53.,R1          ; DO LOOP
6435 042352 012701 000065
6436          ;
6437          ;SETUP TBUF FOR DATA COMPARE
6438 042356 010305          100$: MOV     R3,R5          ; R5 POINTS TO ADDRESS
6439 042360 004737 031446'          JSR     PC,LDBUFC          ; LOAD TBUF WITH ADDRESS DATA PATTERN
6440          ;
6441          ;CLEAR RBUF
6442          JSR     PC,CLRCV          ; CLEAR RECEIVE BUFFER
6443 042364 004737 030236'
6444          ;
6445          ;DUMP INTERNAL MEMORY INTO RBUF
6446 042370 012705 012664'          MOV     #DMPMEM,R5          ; DEFAULT DUMP INTERNAL MEMORY
6447 042374 004737 033256'          JSR     PC,SRWRAM          ; LOAD PCBB AND UDBB
6448          ;
6449 042400 012777 004100 135616 115$: MOV     #DNI!INTE,@PCSR0          ; PRECONDITION INTR EN.
6450 042406 112777 000102 135610          MOV     #INTE!GETCMD,@PCSR0          ; ISSUE GET COMMAND PORT COMMAND
6451 042414 004737 026550'          JSR     PC,CHKDNI          ; DNI ?
6452 042420 103010          BCC     120$          ; YES
6453 042422          FTL
        042422 004737 026652'          JSR     PC,CHKFTL          ; 'FATL' BIT SET?
6454 042426          ERRHRD 055.,ERR010,MSG003          ; NO, REPORT ERROR
        042426 104456          TRAP      C$ERHRD
        042430 000067          .WORD  55
        042432 023331'          .WORD  ERR010
        042434 021736'          .WORD  MSG003
6455 042436          ESCAPE TST          ; AND ABORT TEST
    
```

```

        042436 104410
        042440 000516
        6456
        6457 042442 004737 03026\
        6458
        6459 042446 103010
        6460 042450
                042450 004737 026652'
        6461 042454
                042454 104456
                042456 000070
                042460 023030'
                042462 021736'
        6462 042464
                042464 104410
                042466 000470
        6463
        6464
        6465
        6466 042470 022701 000001
        6467 042474 001003
        6468 042476 012705 000500
        6469 042502 000402
        6470
        6471 042504 012705 002000
        6472 042510 004737 030772'
        6473 042514 103010
        6474 042516
                042516 004737 026652'
        6475 042522
                042522 104456
                042524 000071
                042526 025743'
                042530 022346'
        6476 042532
                042532 104410
                042534 000422
        6477
        6478 042536
        6479 042536 005723
        6480 042540 005301
        6481 042542 001305
        6482
        6483
        6484
        6485
        6486
        6487 042544 005037 016606'
        6488 042550 012703 015712'
        6489 042554 012701 000065
        6490
        6491
        6492 042560 010305
    
```

```

;
;120$: JSR PC,CLRDNI ; WRITE ONE TO CLEAR DNI
; ERROR ?
; NO
        BCC 130$
        FTL
        JSR PC,CHKFTL ; 'FATL' BIT SET?
        ERRHRD 056.,ERR006,MSG003 ; YES, REPORT ERROR
        TRAP C$ERHRD
        .WORD 56
        .WORD ERR006
        .WORD MSG003
        ESCAPE TST ; AND ABORT TEST
        TRAP C$ESCAPE
        .WORD L10042-.
;
;COMPARE RBUF WITH TBUF
;
;130$: CMP #1,R1 ; IS THIS THE LAST 1K BLOCK ?
        BNE 135$ ; NO
        MOV #500,R5 ; YES, ONLY COMPARE 500 WORDS
        BR 136$
;
;135$: MOV #1024.,R5 ; COMPARE 1024. WORDS OF DATA
;136$: JSR PC,CMPMEM ; DATA COMPARE ERROR ?
        BCC 140$ ; NO
        FTL
        JSR PC,CHKFTL ; 'FATL' BIT SET?
        ERRHRD 057.,ERR041,MSG007 ; YES, REPORT ERROR
        TRAP C$ERHRD
        .WORD 57
        .WORD ERR041
        .WORD MSG007
        ESCAPE TST ; AND ABORT TEST
        TRAP C$ESCAPE
        .WORD L10042-.
;
;140$: TST (R3)+ ; BUMP UP TABLE POINTER
        DEC R1 ; DONE 103 READS ?
        BNE 100$
;
;REPEAT TEST WITH COMPLIMENTED DATA PATTERN
;
;WRITE INTERNAL MEMORY WITH DATA = COMPLIMENT OF ADDRESS BY 1K BLOCKS
        CLR EAFLAG ; CLEAR EXT ADDR BITS FLAG
        MOV #MEM13A,R3 ; R3 POINTS TO LINK MEM ADDRESS TABLE
        MOV #53.,R1 ; DO LOOP
;
;WRITE RBUF WITH DATA = ADDRESS
;160$: MOV R3,R5 ; R5 POINTS TO ADDRESS
    
```

```

6493 042562 004737 031446'      JSR      PC,LDBUFC      ; LOAD TBUF WITH COMPLIMENTED DATA
6494                               ;
6495                               ;LOAD INTERNAL RAM MEMORY
6496 042566 012705 012674'      MOV      @LDMEM,R5      ; DEFAULT LOAD INTERNAL MEMORY
6497 042572 004737 033256'      JSR      PC,SRWRAM      ; LOAD PCBB AND UDBB
6498                               ;
6499 042576 012777 004100 135420 ;165$: MOV      @DNI!INTE,@PCSR0  ; PRECONDITION INTR EN.
6500 042604 112777 000102 135412 ; MOVB    @INTE!GETCMD,@PCSR0 ; ISSUE GET COMMAND PORT COMMAND
6501 042612 004737 026550'      JSR      PC,CHKDNI      ; DNI ?
6502 042616 103010                BCC     170$           ; YES
6503 042620                FTL
                                ;
                                ; 'FATL' BIT SET?
                                ; NO, REPORT ERROR
                                TRAP    C$ERHRD
                                .WORD   60
                                .WORD   ERR010
                                .WORD   MSG003
6504 042620 004737 026652'      JSR      PC,CHKFTL
                                ;
                                ; AND ABORT TEST
                                TRAP    C$ESCAPE
                                .WORD   L10042-.
6504 042624                ERRHRD 060.,ERR010,MSG003
6504 042624 104456                ;
6504 042626 000074                ;
6504 042630 023331'                ;
6504 042632 021736'                ;
6505 042634                ESCAPE TST
6505 042634 104410                ;
6505 042636 000320                ;
6506                               ;
6507 042640 004737 030264'      ;170$: JSR      PC,CLRDN1  ; WRITE ONE TO CLEAR DNI
6508                               ; ERROR ?
6509 042644 103010                BCC     180$           ; NO
6510 042646                FTL
                                ;
                                ; 'FATL' BIT SET?
                                ; YES, REPORT ERROR
                                TRAP    C$ERHRD
                                .WORD   61
                                .WORD   ERR006
                                .WORD   MSG003
6511 042646 004737 026652'      JSR      PC,CHKFTL
6511 042652                ERRHRD 061.,ERR006,MSG003
6511 042652 104456                ;
6511 042654 000075                ;
6511 042656 023030'                ;
6511 042660 021736'                ;
6512 042662                ESCAPE TST
6512 042662 104410                ;
6512 042664 000272                ;
6513 042666                ;180$:
6514 042666 005301                DEC     R1              ; DONE 16 WRITES ?
6515 042670 001333                BNE     160$           ; NO
6516                               ;
6517                               ;READ INT RAM MEMORY BY 1K BLOCKS AND COMPARE DATA
6518                               ;
6519 042672 005037 016606'      CLR     EAFLAG          ; CLEAR EXT ADDR BITS FLAG
6520 042676 012703 015712'      MOV     @MEM13A,R3      ; R3 POINTS TO LINK MEM ADDRESS TABLE
6521 042702 012701 000065'      MOV     @53.,R1         ; DO LOOP
6522                               ;
6523                               ;SETUP TBUF FOR DATA COMPARE
6524 042706 010305                ;200$: MOV     R3,R5      ; R5 POINTS TO ADDRESS
6525 042710 004737 031446'      JSR     PC,LDBUFC      ; LOAD TBUF WITH COMPLIMENTED DATA
6526                               ;
6527                               ;CLEAR RBUF
6528 042714 012704 006440'      MOV     @RBUF,R4        ; CLEAR RBUF
6529 042720 012700 002000'      MOV     @1024.,R0
6530 042724 005024                ;210$: CLR     (R4)+
6531 042726 077002                SOB     R0,210$
    
```

```

6532
6533          ;DUMP INTERNAL RAM MEMORY INTO RBUF
6534 042730 012705 012664'      MOV      #DMPMEM,R5      ; DEFAULT DUMP INTERNAL MEMORY
6535 042734 004737 033256'      JSR      PC,SRWRAM      ; LOAD PCBB AND UDB
6536
6537 042740 012777 004100 135256 215$:  MOV      #DNI!INTE,@PCSR0      ; PRECONDITION INTR EN.
6538 042746 112777 000102 135250      MOVB     #INTE!GETCMD,@PCSR0      ; ISSUE GET COMMAND PORT COMMAND
6539 042754 004737 026550'      JSR      PC,CHKDNI      ; DNI ?
6540 042760 103010                BCC      220$           ; YES
6541 042762                FTL
                                JSR      PC,CHKFTL      ; 'FATL' BIT SET?
                                ERRHRD   062.,ERR010,MSG003      ; NO, REPORT ERROR
6542 042766                TRAP      C$ERHRD
                                .WORD   62
                                042766 104456
                                042770 000076
                                042772 023331'
                                042774 021736'
                                .WORD   ERR010
                                .WORD   MSG003
6543 042776                ESCAPE   TST      ; AND ABORT TEST
                                TRAP      C$ESCAPE
                                .WORD   L10042-.
                                042776 104410
                                043000 000156
6544
6545 043002 004737 030264'      220$:  JSR      PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
6546
6547 043006 103010                BCC      230$           ; ERROR ?
6548 043010                FTL
                                JSR      PC,CHKFTL      ; 'FATL' BIT SET?
                                ERRHRD   063.,ERR006,MSG003      ; YES, REPORT ERROR
6549 043014                TRAP      C$ERHRD
                                .WORD   63
                                043014 104456
                                043016 000077
                                043020 023030'
                                043022 021736'
                                .WORD   ERR006
                                .WORD   MSG003
6550 043024                ESCAPE   TST      ; AND ABORT TEST
                                TRAP      C$ESCAPE
                                .WORD   L10042-.
                                043024 104410
                                043026 000130
6551
6552          ;COMPARE RBUF WITH TBUF
6553
6554 043030 022701 000001      230$:  CMP      #1,R1      ; IS THIS THE LAST 1K BLOCK ?
6555 043034 001003                BNE      235$           ; NO
6556 043036 012705 000500      MOV      #500,R5      ; YES, ONLY COMPARE 500 WORDS
6557 043042 000402                BR       236$
6558
6559 043044 012705 002000      235$:  MOV      #1024.,R5      ; COMPARE 1024. WORDS OF DATA
6560 043050 004737 030772'      236$:  JSR      PC,CMEM      ; DATA COMPARE ERROR ?
6561 043054 103010                BCC      240$           ; NO
6562 043056                FTL
                                JSR      PC,CHKFTL      ; 'FATL' BIT SET?
                                ERRHRD   064.,ERR041,MSG007      ; YES, REPORT ERROR
6563 043062                TRAP      C$ERHRD
                                .WORD   64
                                043062 104456
                                043064 000100
                                043066 025743'
                                043070 022346'
                                .WORD   ERR041
                                .WORD   MSG007
    
```

```
6564 043072          ESCAPE TST          ; AND ABORT TEST
      043072 104410
      043074 000062          TRAP      C#ESCAPE
                          .WORD      L10042-.

6565          ;
6566 043076          ; 240#
6567 043076 005723          TST      (R3)+          ; BUMP UP TABLE POINTER
6568 043100 005301          DEC      R1              ; DONE 103 READS ?
6569 043102 001301          BNE     200#

6570          ;
6571 043104 004737 030346'   JSR     PC,CLINTR       ; INSURE DELUA INTR BITS CLEARED
6572          ;
6573 043110          EXIT TST
      043110 104432          TRAP      C#EXIT
      043112 000044          .WORD      L10042-.

6574          ;
6575          ;LOCAL TEST MESSAGE
6576          ;
6577 043114 104 105 114 T12ID: .ASCIZ 'DELUA READ/WRITE INTERNAL MEMORY '
      043117 125 101 040
      043122 122 105 101
      043125 104 057 127
      043130 122 111 124
      043133 105 040 111
      043136 116 124 105
      043141 122 116 101
      043144 114 040 115
      043147 105 115 117
      043152 122 131 040
      043155 000

6578          .EVEN
6579          ;
6580 043156          ENDTST
      043156          L10042: TRAP      C#ETST
      043156 104401
```

6582  
 6583  
 6584  
 6585  
 6586  
 6587  
 6588  
 6589  
 6590  
 6591  
 6592  
 6593  
 6594  
 6595  
 6596  
 6597  
 6598  
 6599

.SBTTL TEST 13: INTERNAL LOOPBACK TEST

```

*****
:
: THIS TEST VERIFIES THAT AN INTERNAL LOOPBACK OPERATION
: CAN BE PERFORMED SUCCESSFULLY.
:
: TEST SEQUENCE:
: 1. WRITE MODE REGISTER = INTERNAL LOOPBACK, PRO 1 MODE
: 2. WRITE RING FORMAT
: 3. WRITE PHYSICAL ADDRESS
: 4. SET UP RINGS AND BUFFERS
: 5. ISSUE START
: 6. CHECK FOR ERRORS
: 7. ISSUE STOP
:
*****
    
```

6600 043160  
 043160  
 6601  
 6602 043160

BGNTST

T13: :

PNTMAC T13ID

043160 012704 044540'  
 043164 004737 032734'

MOV #T13ID,R4  
 JSR PC,PNTID

;GET POINTER TO TEST NAME MESSAGE  
 ;PRINT TEST NUMBER AND NAME

; END OF MACRO EXPANSION OF 'PNTMAC'

6603 043170 004737 033434'  
 6604 043174 103034  
 6605 043176 012777 004100 135020  
 6606 043204 112777 000140 135012  
 6607 043212 004737 027736'  
 6608 043216 103010  
 6609 043220

JSR PC,TINIT  
 BCC 30\$  
 MOV #DNI!INTE,@PCSR0  
 MOVB #INTE!RSET,@PCSR0  
 JSR PC,CKDNI  
 BCC 20\$  
 FTL

; IS A DEVICE RESET NEEDED?  
 ; NO  
 ; PRECONDITION INTR EN.  
 ; YES, RESET DELUA  
 ; DNI ?  
 ; YES

043220 004737 026652'

JSR PC,CHKFTL

; 'FATL' BIT SET?

6610 043224  
 043224 104456  
 043226 000101  
 043230 023030'  
 043232 021736'

ERRHRD 065.,ERR006,MSG003

; NO, REPORT ERROR

TRAP C\$ERHRD  
 .WORD 65  
 .WORD ERR006  
 .WORD MSG003

6611 043234  
 043234 104410  
 043236 001334

ESCAPE TST

; AND ABORT TEST

TRAP C\$ESCAPE  
 .WORD L10043 .

6612  
 6613 043240 004737 030264'  
 6614  
 6615 043244 103010  
 6616 043246

; 20\$: JSR PC,CLR DNI

; WRITE ONE TO CLEAR DNI  
 ; ERROR ?  
 ; NO

043246 004737 026652'

JSR PC,CHKFTL

; 'FATL' BIT SET?

6617 043252  
 043252 104456  
 043254 000102

ERRHRD 066.,ERR006,MSG003

; YES, REPORT ERROR

TRAP C\$ERHRD  
 .WORD 66

```

        043256 023030'
        043260 021736
    6618 043262          ESCAPE TST          ; AND ABORT TEST          .WORD  ERR006
        043262 104410          ;                               .WORD  MSG003
        043264 001306          ;                               TRAP   C$ESCAPE
        ;                               .WORD  L10043 .
    6619
    6620 043266          ; 30$:
    6621 043266 004737 030212' JSR    PC,CLRBUF          ; CLEAR XMIT,RCV BUFFERS
    6622 043272 004737 031732' JSR    PC,LDDFLT         ; LOAD DEFAULT PHY.ADDRESS TABLES
    6623 043276 004737 032032' JSR    PC,LDPCSR         ; ADDRESS OF PCBB -> PCSR2!3
    6624 043302 012777 004100 134714 MOV    #DNI!INTE,@PCSR0  ; PRECONDITION INTR EN.
    6625 043310 112777 000101 134706 MOVB   #INTE!GETPCB,@PCSR0 ; ISSUE GET_PCBB PORT COMMAND
    6626 043316 004737 026550' JSR    PC,CHKDNI         ; DNI?
    6627 043322 103010          BCC    40$              ; YES
    6628 043324          FTL
        043324 004737 026652' JSR    PC,CHKFTL         ; 'FATL' BIT SET?
    6629 043330          ERRHRD 067.,ERR009,MSG003 ; NO, REPORT ERROR
        043330 104456          ;                               TRAP   C$ERHRD
        043332 000103          ;                               .WORD  67
        043334 023245'        ;                               .WORD  ERR009
        043336 021736'        ;                               .WORD  MSG003
    6630 043340          ESCAPE TST          ; AND ABORT TEST          TRAP   C$ESCAPE
        043340 104410          ;                               .WORD  L10043-.
        043342 001230          ;
    6631
    6632 043344 004737 030264' ; 40$: SR    PC,CLRDMI         ; WRITE ONE TO CLEAR DNI
    6633          ;                               ; ERROR ?
    6634 043350 103010          BCC    50$              ; NO
    6635 043352          FTL
        043352 004737 026652' JSR    PC,CHKFTL         ; 'FATL' BIT SET?
    6636 043356          ERRHRD 070.,ERR006,MSG003 ; YES, REPORT ERROR
        043356 104456          ;                               TRAP   C$ERHRD
        043360 000106          ;                               .WORD  70
        043362 023030'        ;                               .WORD  ERR006
        043364 021736'        ;                               .WORD  MSG003
    6637 043366          ESCAPE TST          ; AND ABORT TEST          TRAP   C$ESCAPE
        043366 104410          ;                               .WORD  L10043-.
        043370 001202          ;
    6638          ;WRITE MODE REGISTER = INTERNAL LOOPBACK AND PROM MODE
    6639
    6640          ; 50$:
    6641 043372 012705 012600' MOV    #WMODE,R5         ; DEFAULT WRITE MODE FUNCTION
    6642 043376 004737 032002' JSR    PC,LDPCCB         ; LOAD FUNCTION -> PCBB
    6643 043402 012777 004100 134614 MOV    #DNI!INTE,@PCSR0  ; PRECONDITION INTR EN.
    6644 043410 112777 000102 134606 MOVB   #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
    6645 043416 004737 026550' JSR    PC,CHKDNI         ; DNI ?
    6646 043422 103010          BCC    60$              ; YES
    6647 043424          FTL
        043424 004737 026652' JSR    PC,CHKFTL         ; 'FATL' BIT SET?
    6648 043430          ERRHRD 071.,ERR010,MSG003 ; NO, REPORT ERROR
        043430 104456          ;                               TRAP   C$ERHRD
    
```

```

        043432 000107                                .WORD 71
        043434 023331'                                .WORD ERR010
        043436 021736'                                .WORD MSG003
6649 043440                ESCAPE TST                ; AND ABORT TEST
        043440 104410                                TRAP C$ESCAPE
        043442 001130                                .WORD L10043-.
6650
6651 043444 004737 030264'    ;
6652                                ;
6653 043450 103010                BCC 70$                ; WRITE ONE TO CLEAR DNI
6654 043452                FTL                                ; ERROR ?
                                                ; NO
        043452 004737 026652'    JSR PC,CLRDNI          ; 'FATL' BIT SET?
6655 043456                ERRHRD 072.,ERR006,MSG003    ; YES, REPORT ERROR
        043456 104456                                TRAP C$ERHRD
        043460 000110                                .WORD 72
        043462 023030'                                .WORD ERR006
        043464 021736'                                .WORD MSG003
6656 043466                ESCAPE TST                ; AND ABORT TEST
        043466 104410                                TRAP C$ESCAPE
        043470 001102                                .WORD L10043-.
6657
6658                                ;WRITE RING FORMAT
6659                                ;
6660 043472 012705 012540'    70$: MOV $WTRNGS,R5    ; DEFAULT WRITE RING FORMAT FUNCTION
6661 043476 004737 032002'    JSR PC,LDPCCB        ; LOAD FUNCTION -> PCBB
6662 043502 012705 012704'    MOV $RFRMT,R5       ; DEFAULT RING FORMAT
6663 043506 012700 000006    MOV $6,R0            ; FORMAT = SIX WORDS
6664 043512 004737 032260'    JSR PC,LDUDBB       ; LOAD RING FORMAT -> UDBB
6665 043516 012777 004100 134500    MOV $DNI!INTE,@PCSR0 ; PRECONDITION INTR EN.
6666 043524 112777 000102 134472    MOV $INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
6667 043532 004737 026550'    JSR PC,CHKDNI       ; DNI ?
6668 043536 103010                BCC 80$                ; YES
6669 043540                FTL
        043540 004737 026652'    JSR PC,CHKFTL       ; 'FATL' BIT SET?
6670 043544                ERRHRD 073.,ERR010,MSG003    ; NO, REPORT ERROR
        043544 104456                                TRAP C$ERHRD
        043546 000111                                .WORD 73
        043550 023331'                                .WORD ERR010
        043552 021736'                                .WORD MSG003
6671 043554                ESCAPE TST                ; AND ABORT TEST
        043554 104410                                TRAP C$ESCAPE
        043556 001014                                .WORD L10043-.
6672
6673 043560 004737 030264'    ;
6674                                ;
6 5 043564 103010                BCC 90$                ; WRITE ONE TO CLEAR DNI
6676 043566                FTL                                ; ERROR ?
                                                ; NO
        043566 004737 026652'    JSR PC,CLRDNI          ; 'FATL' BIT SET?
6677 043572                ERRHRD 074.,ERR006,MSG003    ; YES, REPORT ERROR
        043572 104456                                TRAP C$ERHRD
        043574 000112                                .WORD 74
    
```



```

043576 023030'
043600 021736'
6678 043602          ESCAPE TST          ; AND ABORT TEST          .WORD  ERR006
043602          104410          .WORD  MSG003
043604          000766          TRAP   C$ESCAPE
                                .WORD  L10043 .

6679          ;
6680          ;WRITE PHYSICAL ADDRESS
6681
6682 043606          90$:
6683 043606 012705 000272'          MOV    #DEFAULT,R5          ; GET DEFAULT PHYSICAL ADDRESS
6684 043612 004737 032050'          JSR    PC,LDPHYA          ; PLACE IT IN DATA TABLE
6685 043616 012705 012500'          MOV    #WTPHYA,R5          ; DEFAULT WRITE PHYSICAL ADDR FUNC
6686 043622 004737 032002'          JSR    PC,LDPCBB          ; LOAD FUNCTION -> PCBB
6687 043626 012777 004100 134370          MOV    #DNI!INTE,@PCSRO          ; PRECONDITION INTR EN.
6688 043634 112777 000102 134362          MOVB  #INTE!GETCMD,@PCSRO          ; ISSUE GET_CMD PORT COMMAND
6689 043642 004737 026550'          JSR    PC,CHKDNI          ; DNI ?
6690 043646 103010          BCC   100$          ; YES
6691 043650          FTL

043650 004737 026652'          JSR    PC,CHKFTL          ; 'FATL' BIT SET?

6692 043654          ERRHRD 075.,ERR010,MSG003          ; NO, REPORT ERROR
043654 104456          TRAP   C$ERHRD
043656 000113          .WORD  75
043660 023331'          .WORD  ERR010
043662 021736'          .WORD  MSG003
6693 043664          ESCAPE TST          ; AND ABORT TEST          TRAP   C$ESCAPE
043664 104410          .WORD  L10043-.
043666 000704

6694          ;
6695 043670 004737 030264'          100$: JSR    PC,CLR DNI          ; WRITE ONE TO CLEAR DNI
6696          ; ERROR ?
6697 043674 103010          BCC   110$          ; NO
6698 043676          FTL

043676 004737 026652'          JSR    PC,CHKFTL          ; 'FATL' BIT SET?

6699 043702          ERRHRD 076.,ERR006,MSG003          ; YES, REPORT ERROR
043702 104456          TRAP   C$ERHRD
043704 000114          .WORD  76
043706 023030'          .WORD  ERR006
043710 021736'          .WORD  MSG003
6700 043712          ESCAPE TST          ; AND ABORT TEST          TRAP   C$ESCAPE
043712 104410          .WORD  L10043-.
043714 000656

6701          ;
6702          ;SET UP RINGS FOR ONE BUFFER LOOPBACK
6703
6704 043716 012705 014410'          110$: MOV    #TDRB1A,R5          ; DEFAULT ONE BUFFER TRANSMIT RING
6705 043722 004737 032164'          JSR    PC,L0TDRB          ; LOAD TDRB
6706 043726 012705 012750'          MOV    #RDRB1A,R5          ; DEFAULT ONE BUFFER RECEIVE RING
6707 043732 004737 032070'          JSR    PC,LDRDRB          ; LOAD RDRB
6708          ;
6709          ;SET UP BUFFERS AND START
6710
6711 043736 005037 016562'          CLR    DOCRC          ; NO APPEND CRC
6712 043742 012737 000006 016560'          MOV    #6,BYTCNT          ; DATA BYTE COUNT
    
```

6713	043750	004737	033006'		JSR	PC,SETBUF		; SET UP BUFFERS		
6714	043754	012777	004100	134242	MOV	0DNI!INTE,@PCSR0		; PRECONDITION INTR EN.		
6715	043762	112777	000104	134234	MOVB	0INTE!START,@PCSR0		; ISSUE START PORT COMMAND		
6716	043770	004737	026550'		JSR	PC,CHKDNI		; DNI?		
6717	043774	103010			BCC	120\$		; YES		
6718	043776				FTL					
	043776	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
6719	044002				ERRHRD	077.,ERR012,MSG003		; NO, REPORT ERROR		
	044002	104456							TRAP	C\$ERHRD
	044004	000115							.WORD	77
	044006	023447'							.WORD	ERR012
	044010	021736'							.WORD	MSG003
6720	044012				ESCAPE	TST		; AND ABORT TEST		
	044014	104410							TRAP	C\$ESCAPE
	044014	000556							.WORD	L10043-
6721										
6722	044016	004737	030264'	i	JSR	PC,CLRDN1	120\$:	; WRITE ONE TO CLEAR DNI		
6723								; ERROR ?		
6724	044022	103010			BCC	130\$		; NO		
6725	044024				FTL					
	044024	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
6726	044030				ERRHRD	080.,ERR006,MSG003		; YES, REPORT ERROR		
	044030	104456							TRAP	C\$ERHRD
	044032	000120							.WORD	80
	044034	023030'							.WORD	ERR006
	044036	021736'							.WORD	MSG003
6727	044040				ESCAPE	TST		; AND ABORT TEST		
	044040	104410							TRAP	C\$ESCAPE
	044042	000530							.WORD	L10043-
6728										
6729	044044	004737	027566'	i	JSR	PC,CHKTXI	130\$:	; TXI ?		
6730	044050	103010			BCC	140\$		; YES		
6731	044052				FTL					
	044052	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
6732	044056				ERRHRD	081.,ERR013,MSG003		; NO, REPORT ERROR		
	044056	104456							TRAP	C\$ERHRD
	044060	000121							.WORD	81
	044062	023530'							.WORD	ERR013
	044064	021736'							.WORD	MSG003
6733	044066				ESCAPE	TST		; AND ABORT TEST		
	044066	104410							TRAP	C\$ESCAPE
	044070	000502							.WORD	L10043-
6734										
6735	044072	004737	030500'	i	JSR	PC,CLR TXI	140\$:	; WRITE ONE TO CLEAR TXI		
6736								; ERROR ?		
6737	044076	103010			BCC	150\$		; NO		
6738	044100				FTL					
	044100	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
6739	044104				ERRHRD	082.,ERR014,MSG003		; YES, REPORT ERROR		



```

6764 044242 004737 030432'      180$: JSR    PC,CLRRXI      ; WRITE ONE TO CLEAR RXI
6765                                ; ERROR ?
6766 044246 103010              BCC    190$              ; NO
6767 044250                                FTL

        044250 004737 026652'      JSR    PC,CHKFTL        ; 'FATL' BIT SET?
6768 044254                                ERRHRD 086.,ERR016,MSG003 ; YES, REPORT ERROR
        044254 104456                                TRAP   C$ERHRD
        044256 000126                                .WORD 86
        044260 023660'                                .WORD ERR016
        044262 021736'                                .WORD MSG003
6769 044264                                ESCAPE TST              ; AND ABORT TEST
        044264 104410                                TRAP   C$ESCAPE
        044266 000304                                .WORD L10043-.
6770                                ;
6771 044270 012705 000660'      190$: MOV    #RDRB,R5      ; CHECK RDRB OWNERSHIP
6772 044274 004737 027024'      JSR    PC,CHKOWN       ; OWN = PORT DRIVER ?
6773 044300 103010              BCC    200$              ; YES
6774 044302                                FTL

        044302 004737 026652'      JSR    PC,CHKFTL        ; 'FATL' BIT SET?
6775 044306                                ERRHRD 087.,ERR017      ; NO, REPORT ERROR?
        044306 104456                                TRAP   C$ERHRD
        044310 000127                                .WORD 87
        044312 023726'                                .WORD ERR017
        044314 000000                                .WORD 0
6776 044316                                ESCAPE TST              ; AND ABORT TEST
        044316 104410                                TRAP   C$ESCAPE
        044320 000252                                .WORD L10043-.
6777                                ;
6778 044322 012705 016454'      200$: MOV    #RDR20C,R5  ; POINT TO EXPECTED RDRB
6779 044326 004737 032340'      JSR    PC,LDXRDR       ; LOAD INTO XRDRBO TABLE
6780 044332 012705 000660'      MOV    #RDRB,R5      ; CHECK RDRB
6781 044336 004737 027206'      JSR    PC,CHKRDR       ; ERRORS ?
6782 044342 103010              BCC    210$              ; NO
6783 044344                                FTL

        044344 004737 026652'      JSR    PC,CHKFTL        ; 'FATL' BIT SET?
6784 044350                                ERRHRD 090.,ERR021,MSG006 ; YES, REPORT ERROR
        044350 104456                                TRAP   C$ERHRD
        044352 000132                                .WORD 90
        044354 024267'                                .WORD ERR021
        044356 022204'                                .WORD MSG006
6785 044360                                ESCAPE TST              ; AND ABORT TEST
        044360 104410                                TRAP   C$ESCAPE
        044362 000210                                .WORD L10043-.
6786                                ;
6787                                ;COMPARE RBUF WITH TBUF
6788                                ;
6789 044364 013705 016560'      210$: MOV    BYTCNT,R5  ; NUMBER OF DATA COMPARES
6790 044370 004737 030712'      JSR    PC,CMPDAT       ; DATA COMPARE ERROR ?
6791 044374 103006              BCC    220$              ; NO
6792 044376                                ERRHRD 091.,ERR022,MSG007 ; YES, REPORT ERROR
        044376 104456                                TRAP   C$ERHRD
    
```

	044400	000133						.WORD	91
	044402	024350'						.WORD	ERR022
	044404	022346'						.WORD	MSG007
6793	044406		ESCAPE	TST					
	044406	104410						TRAP	C\$ESCAPE
	044410	000162						.WORD	L10043 .
6794									
6795	044412								
6796	044412	012705	006472'	MOV	#RBUF+32,R5				
6797									
6798	044416	004737	030642'	JSR	PC,CMPCRC				
6799	044422	103006		BCC	230\$				
6800	044424			ERRHRD	092.,ERR023,MSG008				
	044424	104456						TRAP	C\$ERHRD
	044426	000134						.WORD	92
	C44430	024417'						.WORD	ERR023
	044432	022400'						.WORD	MSG008
6801	044434		ESCAPE	TST					
	044434	104410						TRAP	C\$ESCAPE
	044436	000134						.WORD	L10043-.
6802									
6803	044440	012777	0C4100	133556	230\$:	MOV	#DNI!INTE,@PCSRC		
6804	044446	112777	000117	133550		MOV	#INTE!STOP,@PCSR0		
6805	044454	004737	026550'			JSR	PC,CHKDNI		
6806	044460	103010				BCC	240\$		
6807	044462					FTL			
	044462	004737	026652'			JSR	PC,CHKFTL		
6808	044466					ERRHRD	093.,ERR019,MSG003		
	044466	104456							
	044470	000135						TRAP	C\$ERHRD
	044472	024126'						.WORD	93
	044474	021736'						.WORD	ERR019
6809	044476		ESCAPE	TST				.WORD	MSG003
	044476	104410							
	044500	000072						TRAP	C\$ESCAPE
6810								.WORD	L10043-.
6811	044502	004737	030264'	240\$:	JSR	PC,CLRDNI			
6812									
6813	044506	103010				BCC	250\$		
6814	044510					FTL			
	044510	004737	026652'			JSR	PC,CHKFTL		
6815	044514					ERRHRD	094.,ERR006,MSG003		
	044514	104456							
	044516	000136						TRAP	C\$ERHRD
	044520	023030'						.WORD	94
	044522	021736'						.WORD	ERR006
6816	044524		ESCAPE	TST				.WORD	MSG003
	044524	104410							
	044526	000044						TRAP	C\$ESCAPE
6817	044530							.WORD	L10043-.
6818	044530	004737	030346'	250\$:	JSR	PC,CLINTR			
6819									
6820	044534		EXIT	TST					

TRAP C#EXIT  
.WORD L10043-

```
044534 104432
044536 000034
6821
6822 ;LOCAL TEST MESSAGE
6823
6824 044540 104 105 114 T13ID:.ASCIZ 'DELUA INTERNAL LOOPBACK '
044543 125 101 040
044546 111 116 124
044551 105 122 116
044554 101 114 040
044557 114 117 117
044562 120 102 101
044565 103 113 040
044570 000
6825 .EVEN
6826
6827 044572 ENDTST
044572
044572 104401
```

L10043: TRAP C#ETST



```

044662 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
6869 044666                ERRHRD   096.,ERR006,MSG003 ; YES, REPORT ERROR
      044666 104456                TRAP      C$ERHRD
      044670 000140                .WORD    96
      044672 023030'                .WORD    ERR006
      044674 021736'                .WORD    MSG003
6870 044676                ESCAPE   TST                ; AND ABORT TEST
      044676 104410                TRAP      C$ESCAPE
      044700 001254                .WORD    L10044-.
6871
6872 044702                ;
6873 044702 004737 030212'      JSR      PC,CLRBUF          ; CLEAR XMIT,RCV BUFFERS
6874 044706 004737 031732'      JSR      PC,LDDFLT          ; LOAD DEF PHY.ADDRESS TABLES
6875 044712 004737 032032'      JSR      PC,LDPCSR          ; ADDRESS OF PCBB -> PCSR2!3
6876 044716 012777 004100 133300  MOV      #DNI!INTE,@PCSR0   ; PRECONDITION INTR EN.
6877 044724 112777 000101 133272  MOVB    #INTE!GETPCB,@PCSR0 ; ISSUE GET_PCBB PORT COMMAND
6878 044732 004737 026550'      JSR      PC,CHKDNI          ; DNI?
6879 044736 103010                BCC      40$                ; YES
6880 044740                FTL
044740 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
6881 044744                ERRHRD   097.,ERR009,MSG003 ; NO, REPORT ERROR
      044744 104456                TRAP      C$ERHRD
      044746 000141                .WORD    97
      044750 023245'                .WORD    ERR009
      044752 021736'                .WORD    MSG003
6882 044754                ESCAPE   TST                ; AND ABORT TEST
      044754 104410                TRAP      C$ESCAPE
      044756 001176                .WORD    L10044-.
6883
6884 044760 004737 030264'      JSR      PC,CLRDNI          ; WRITE ONE TO CLEAR DNI
6885                                ; ERROR ?
6886 044764 103010                BCC      50$                ; NO
6887 044766                FTL
044766 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
6888 044772                ERRHRD   100.,ERR006,MSG003 ; YES, REPORT ERROR
      044772 104456                TRAP      C$ERHRD
      044774 000144                .WORD    100
      044776 023030'                .WORD    ERR006
      045000 021736'                .WORD    MSG003
6889 045002                ESCAPE   TST                ; AND ABORT TEST
      045002 104410                TRAP      C$ESCAPE
      045004 001150                .WORD    L10044-.
6890
6891                                ;WRITE MODE REGISTER = INTERNAL LOOPBACK, PROM AND DISABLE XMIT CRC MODE
6892
6893 045006 012705 012620'      50$:   MOV      #WTMOD2,R5      ; WRITE MODE FUNCTION, DISABLE
      6894                                ; TRANSMIT CRC
6895 045012 004737 032002'      JSR      PC,LDPCBB          ; LOAD FUNCTION -> PCBB
6896 045016 012777 004100 133200  MOV      #DNI!INTE,@PCSR0   ; PRECONDITION INTR EN.
6897 045024 112777 000102 133172  MOVB    #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
6898 045032 004737 026550'      JSR      PC,CHKDNI          ; DNI ?
6899 045036 103010                BCC      60$                ; YES

```



```

6900 045040          FTL
      045040 004737 026652'          JSR    PC,CHKFTL          ; 'FATL' BIT SET?
6901 045044          ERRHRD 101.,ERR010,MSG003          ; NO, REPORT ERROR
      045044 104456          TRAP    C$ERHRD
      045046 000145          .WORD  101
      045050 023331'          .WORD  ERR010
      045052 021736'          .WORD  MSG003
6902 045054          ESCAPE TST          ; AND ABORT TEST
      045054 104410          TRAP    C$ESCAPE
      045056 001076          .WORD  L10044-.
6903          ;
6904 045060 004737 030264'          ; 60$: JSR    PC,CLRDN1          ; WRITE ONE TO CLEAR DNI
6905          ;
6906 045064 103010          BCC    70$          ; ERROR ?
6907 045066          FTL          ; NO
      045066 004737 026652'          JSR    PC,CHKFTL          ; 'FATL' BIT SET?
6908 045072          ERRHRD 102.,ERR006,MSG003          ; YES, REPORT ERROR
      045072 104456          TRAP    C$ERHRD
      045074 000146          .WORD  102
      045076 023030'          .WORD  ERR006
      045100 021736'          .WORD  MSG003
6909 045102          ESCAPE TST          ; AND ABORT TEST
      045102 104410          TRAP    C$ESCAPE
      045104 001050          .WORD  L10044-.
6910          ;
6911          ;WRITE RING FORMAT
6912          ;
6913 045106 012705 012540'          ; 70$: MOV    #WTRNGS,R5          ; DEFAULT WRITE RING FORMAT FUNCTION
6914 045112 004737 032002'          JSR    PC,LDPCBB          ; LOAD FUNCTION -> PCBB
6915 045116 012705 012704'          MOV    #RFRMT,R5          ; DEFAULT RING FORMAT
6916 045122 012700 000006          MOV    #6,RO          ; FORMAT = SIX WORDS
6917 045126 004737 032260'          JSR    PC,LDUDBB          ; LOAD RING FORMAT -> UDBB
6918 045132 012777 004100 133064          MOV    #DNI!INTE,@PCSR0          ; PRECONDITION INTR EN.
6919 045140 112777 000102 133056          MOVB  #INTE!GETCMD,@PCSR0          ; ISSUE GET_CMD PORT COMMAND
6920 045146 004737 026550'          JSR    PC,CHKDNI          ; DNI ?
6921 045152 103010          BCC    80$          ; YES
6922 045154          FTL
      045154 004737 026652'          JSR    PC,CHKFTL          ; 'FATL' BIT SET?
6923 045160          ERRHRD 103.,ERR010,MSG003          ; NO, REPORT ERROR
      045160 104456          TRAP    C$ERHRD
      045162 000147          .WORD  103
      045164 023331'          .WORD  ERR010
      045166 021736'          .WORD  MSG003
6924 045170          ESCAPE TST          ; AND ABORT TEST
      045170 104410          TRAP    C$ESCAPE
      045172 000762          .WORD  L10044-.
6925          ;
6926 045174 004737 030264'          ; 80$: JSR    PC,CLRDN1          ; WRITE ONE TO CLEAR DNI
6927          ;
6928 045200 103010          BCC    50$          ; ERROR ?
6929 045202          FTL          ; NO
    
```

```

045202 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
6930 045206                ERRHRD   104.,ERR006,MSG003 ; YES, REPORT ERROR
      045206 104456                TRAP      C$ERHRD
      045210 000150                .WORD    104
      045212 023030'                .WORD    ERR006
      045214 021736'                .WORD    MSG003
6931 045216                ESCAPE   TST              ; AND ABORT TEST
      045216 104410                TRAP      C$ESCAPE
      045220 000734                .WORD    L10044 .
6932
6933                ;WRITE PHYSICAL ADDRESS
6934
6935 045222                90$:
6936 045222 012705 000272'      MOV      @DEFAULT,R5      ; GET DEFAULT PHYSICAL ADDRESS
6937 045226 004737 032050'      JSR      PC,LDPHYA        ; STORE IT IN DEFAULT TABLE
6938 045232 012705 012500'      MOV      @WTPHYA,R5      ; DEFAULT WRITE PHYSICAL ADDR FUNC
6939 045236 004737 032002'      JSR      PC,LDPCBB        ; LOAD FUNCTION -> PCBB
6940 045242 012777 004100 132754  MOV      @DNI!INTE,@PCSR0 ; PRECONDITION INTR EN.
6941 045250 112777 000102 132746  MOVB     @INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
6942 045256 004737 026550'      JSR      PC,CHKDNI        ; DNI ?
6943 045262 103010                BCC      100$             ; YES
6944 045264                FTL
045264 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
6945 045270                ERRHRD   105.,ERR010,MSG003 ; N7, REPORT ERROR
      045270 104456                TRAP      C$ERHRD
      045272 000151                .WORD    105
      045274 023331'                .WORD    ERR010
      045276 021736'                .WORD    MSG003
6946 045300                ESCAPE   TST              ; AND ABORT TEST
      045300 104410                TRAP      C$ESCAPE
      045302 000652                .WORD    L10044-.
6947
6948 045304 004737 030264'      ; 100$: JSR      PC,CLRDN1      ; WRITE ONE TO CLEAR DNI
6949                                ; ERROR ?
6950 045310 103010                BCC      110$             ; NO
6951 045312                FTL
045312 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
6952 045316                ERRHRD   106.,ERR006,MSG003 ; YES, REPORT ERROR
      045316 104456                TRAP      C$ERHRD
      045320 000152                .WORD    106
      045322 023030'                .WORD    ERR006
      045324 021736'                .WORD    MSG003
6953 045326                ESCAPE   TST              ; AND ABORT TEST
      045326 104410                TRAP      C$ESCAPE
      045330 000624                .WORD    L10044-.
6954
6955                ;SET UP RINGS FOR ONE BUFFER LOOPBACK
6956
6957 045332 012705 014450'      110$:  MOV      @TDRB1B,R5      ; DEFAULT ONE BUFFER TRANSMIT RING
6958 045336 004737 032164'      JSR      PC,LDTDRB        ; LOAD TDRB
6959 045342 012705 012750'      MOV      @RDRB1A,R5      ; DEFAULT ONE BUFFER RECEIVE RING

```

```

6960 045346 004737 032070'          JSR    PC,LDRDRB          ; LOAD RDRB
6961
6962                                ;SET UP BUFFERS AND START
6963
6964 045352 012737 000001 016562'    MOV    #1,DOCRC          ; APPEND CRC AND SAVE
6965 045360 012737 000006 016560'    MOV    #6,BYTCNT        ; DATA BYTES/PACKET
6966 045366 004737 033006'          JSR    PC,SETBUF         ; SET UP BUFFERS
6967
6968 045372 012777 004100 132624    MOV    #DNI!INTE,@PCSRO ; PRECONDITION INTR EN.
6969 045400 112777 000104 132616    MOVB   #INTE!START,@PCSRO ; ISSUE START PORT COMMAND
6970 045406 004737 026550'          JSR    PC,CHKDNI         ; DNI?
6971 045412 103010                    BCC    120$              ; YES
6972 045414                    FTL

        045414 004737 026652'          JSR    PC,CHKFTL         ; 'FATL' BIT SET?

6973 045420                    ERRHRD 107.,ERR012,MSG003    ; NO, REPORT ERROR
        045420 104456                    TRAP   C$ERHRD
        045422 000153                    .WORD 107
        045424 023447'                    .WORD ERR012
        045426 021736'                    .WORD MSG003

6974 045430                    ESCAPE TST                ; AND ABORT TEST
        045430 104410                    TRAP   C$ESCAPE
        045432 000522                    .WORD L10044-.

6975
6976 045434 004737 030264'          ;120$: JSR    PC,CLRDN1        ; WRITE ONE TO CLEAR DNI
6977                                ; ERROR ?
6978 045440 103010                    BCC    130$              ; NO
6979 045442                    FTL

        045442 004737 026652'          JSR    PC,CHKFTL         ; 'FATL' BIT SET?

6980 045446                    ERRHRD 110.,ERR006,MSG003    ; YES, REPORT ERROR
        045446 104456                    TRAP   C$ERHRD
        045450 000156                    .WORD 110
        045452 023030'                    .WORD ERR006
        045454 021736'                    .WORD MSG003

6981 045456                    ESCAPE TST                ; AND ABORT TEST
        045456 104410                    TRAP   C$ESCAPE
        045460 000474                    .WORD L10044.

6982
6983 045462 004737 027566'          ;130$: JSR    PC,CHKTXI        ; TXI ?
6984 045466 103010                    BCC    140$              ; YES
6985 045470                    FTL

        045470 004737 026652'          JSR    PC,CHKFTL         ; 'FATL' BIT SET?

6986 045474                    ERRHRD 111.,ERR013,MSG003    ; NO, REPORT ERROR
        045474 104456                    TRAP   C$ERHRD
        045476 000157                    .WORD 111
        045500 023530'                    .WORD ERR013
        045502 021736'                    .WORD MSG003

6987 045504                    ESCAPE TST                ; AND ABORT TEST
        045504 104410                    TRAP   C$ESCAPE
        045506 000446                    .WORD L10044.

6988
6989 045510 004737 030500'          ;140$: JSR    PC,CLRTXI        ; WRITE ONE TO CLEAR TXI
  
```



```

045646 000163 .WORD 115
045650 023627' .WORD ERR015
045652 021736' .WORD MSG003
7016 045654 ESCAPE TST ; AND ABORT TEST
045654 104410 TRAP C$ESCAPE
045656 000276 .WORD L10044-.

7017
7018 045660 004737 030432' ; 180$: JSR PC,CLRRXI ; WRITE ONE TO CLEAR RXI
7019 ; ERROR ?
7020 045664 103010 BCC 190$ ; NO
7021 045666 FTL

045666 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

7022 045672 ERRHRD 116.,ERR016,MSG003 ; YES, REPORT ERROR
045672 104456 TRAP C$ERHRD
045674 000164 .WORD 116
045676 023660' .WORD ERR016
045700 021736' .WORD MSG003
7023 045702 ESCAPE TST ; AND ABORT TEST
045702 104410 TRAP C$ESCAPE
045704 000250 .WORD L10044-.

7024
7025 045706 012705 000660' ; 190$: MOV #RDRB,R5 ; CHECK RDRB OWNERSHIP
7026 045712 004737 027024' JSR PC,CHKOWN ; OWN = PORT DRIVER ?
7027 045716 103010 BCC 200$ ; YES
7028 045720 FTL

045720 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

7029 045724 ERRHRD 117.,ERR017 ; NO, REPORT ERROR
045724 104456 TRAP C$ERHRD
045726 000165 .WORD 117
045730 023726' .WORD ERR017
045732 000000 .WORD 0
7030 045734 ESCAPE TST ; AND ABORT TEST
045734 104410 TRAP C$ESCAPE
045736 000216 .WORD L10044 .

7031
7032 045740 012705 016374' ; 200$: MOV #RDR14B,R5 ; POINT TO EXPECTED RDRB
7033 045744 004737 032340' JSR PC,LDXRDR ; LOAD INTO XRDRBO TABLE
7034 045750 012705 000660' MOV #RDRB,R5 ; CHECK RDRB
7035 045754 004737 027206' JSR PC,CHKRDR ; ERRORS ?
7036 045760 103010 BCC 210$ ; NO
7037 045762 FTL

045762 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

7038 045766 ERRHRD 120.,ERR021,MSG006 ; YES, REPORT ERROR
045766 104456 TRAP C$ERHRD
045770 000170 .WORD 120
045772 024267' .WORD ERR021
045774 022204' .WORD MSG006
7039 045776 ESCAPE TST ; AND ABORT TEST
045776 104410 TRAP C$ESCAPE
046000 000154 .WORD L10044-.

7040
  
```

```

7041                                     ;COMPARE RBUF WITH TBUF
7042
7043 046002 013705 016560'           210$:  MOV    BYTCNT,R5           ; COMPARE DATA
7044 046006 004737 030712'           JSR    PC,CMPCAT           ; DATA COMPARE ERROR ?
7045 046012 103006                   BCC    230$                ; NO
7046 046014                               ERRHRD 121.,ERR022,MSG007   ; YES, REPORT ERROR
                                TRAP   C$ERHRD
                                .WORD  121
                                .WORD  ERR022
                                .WORD  MSG007
                                046014 104456
                                046016 000171
                                046020 024350'
                                046022 022346'
7047 046024                               ESCAPE TST                   ; AND ABORT TEST
                                TRAP   C$ESCAPE
                                .WORD  L10044-.
                                046024 104410
                                046026 000126
7048
7049 046030 012777 004100 132166     ; 230$:  MOV    @DNI!INTE,@PCSRO           ; PRECONDITION INTR EN.
7050 046036 112777 000117 132160     MOVB   @INTE!STOP,@PCSRO   ; ISSUE STOP PORT COMMAND
7051 046044 004737 026550'           JSR    PC,CHKDNI           ; DNI ?
7052 046050 103010                   BCC    240$                ; YES
7053 046052                               FTL
                                JSR    PC,CHKFTL           ; 'FATL' BIT SET?
                                ERRHRD 122.,ERF019,MSG003   ; NO, REPORT ERROR
                                TRAP   C$ERHRD
                                .WORD  122
                                .WORD  ERR019
                                .WORD  MSG003
7054 046056                               ESCAPE TST                   ; AND ABORT TEST
                                TRAP   C$ESCAPE
                                .WORD  L10044-.
                                046056 104456
                                046060 000172
                                046062 024126'
                                046064 021736'
7055 046066                               ESCAPE TST                   ; AND ABORT TEST
                                TRAP   C$ESCAPE
                                .WORD  L10044-.
                                046066 104410
                                046070 000064
7056
7057 046072 004737 030264'           ; 240$:  JSR    PC,CLRDNI           ; WRITE ONE TO CLEAR DNI
7058                               BCC    250$                ; ERROR ?
7059 046076 103010                   FTL                          ; NO
7060 046100                               JSR    PC,CHKFTL           ; 'FATL' BIT SET?
                                ERRHRD 123.,ERR006,MSG003   ; YES, REPORT ERROR
                                TRAP   C$ERHRD
                                .WORD  123
                                .WORD  ERR006
                                .WORD  MSG003
7061 046104                               ESCAPE TST                   ; AND ABORT TEST
                                TRAP   C$ESCAPE
                                .WORD  L10044-.
                                046104 104456
                                046106 000173
                                046110 023030'
                                046112 021736'
7062 046114                               ESCAPE TST                   ; AND ABORT TEST
                                TRAP   C$ESCAPE
                                .WORD  L10044-.
                                046114 104410
                                046116 000036
7063 046120                               ; 250$:  JSR    PC,CLINTR           ; INSURE DELUA INTR BITS CLEAR
7064 046120 004737 030346'           EXIT    TST
7065
7066 046124                               TRAP   C$EXIT
                                046124 104432                .WORD  L10044-.
                                046126 000026
    
```

```
7068 ;LOCAL TEST MESSAGE
7069
7070 046130 104 105 114 T14ID:.ASCIZ 'DELUA CRC CHECKING '
      046133 125 101 040
      046136 103 122 103
      046141 040 103 110
      046144 105 103 113
      046147 111 116 107
      046152 040 000
7071 .EVEN
7072
7073 046154 ENDTST
      046154
      046154 104401
```

L10044: TRAP C\$ETST

7075  
 7076  
 7077  
 7078  
 7079  
 7080  
 7081  
 7082  
 7083  
 7084  
 7085  
 7086  
 7087  
 7088  
 7089  
 7090  
 7091  
 7092  
 7093  
 7094  
 7095  
 7096  
 7097

.SBTTL TEST 15: FORCE CRC ERROR TEST

```

*****
:
: THIS TEST VERIFIES THAT A CRC ERROR CAN BE DETECTED.
: AN INTERNAL LOOPBACK IS PERFORMED WHILE IN
: THE DISABLE TRANSMIT CRC MODE.
: WITH A BAD CRC VALUE APPENDED TO THE TRANSMIT BUFFER
: A CRC ERROR IS EXPECTED IN THE RECEIVE DESCRIPTOR RING.
:
: TEST SEQUENCE:
:   1. WRITE MODE REGISTER = INTERNAL LOOPBACK, PROM,
:                                     and DISABLE TRANSMIT CRC MODE
:   2. WRITE RING FORMAT
:   3. WRITE PHYSICAL ADDRESS
:   4. SET UP RINGS AND BUFFERS
:   5. APPEND BAD CRC VALUE TO TRANSMIT BUFFER
:   6. ISSUE START
:   7. CHECK FOR CRC ERROR IN RDRB+4
:   8. ISSUE STOP
:
*****
  
```

7098 046156  
 046156  
 7099  
 7100 046156

BGNTST

T15::

PNTMAC T15ID

046156 012704 047544'  
 046162 004737 032734'

```

MOV #T15ID,R4 ;GET POINTER TO TEST NAME MESSAGE
JSR PC,PNTID ;PRINT TEST NUMBER AND NAME
  
```

END OF MACRO EXPANSION OF 'PNTMAC'

7101 046166 004737 033434'  
 7102 046172 103034  
 7103 046174 012777 004100 132022  
 7104 046202 112777 000140 132014  
 7105 046210 004737 027736'  
 7106 046214 103010  
 7107 046216

```

JSR PC,TINIT ; IS A DEVICE RESET NEEDED?
BCC 30$ ; NO
MOV #DNI!INTE,@PCSRO ; PRECONDITION INTR EN.
MOVB #INTE!RSET,@PCSRO ; YES, RESET DELUA
JSR PC,CKDNI ; DNI ?
BCC 20$ ; YES
FTL
  
```

046216 004737 026652'

JSR PC,CHKFTL ; 'FATL' BIT SET?

7108 046222  
 046222 104456  
 046224 000174  
 046226 023030'  
 046230 021736'

ERRHRD 124.,ERR006,MSG003 ; NO, REPORT ERROR

```

TRAP C$ERHRD
.WORD 124
.WORD ERR006
.WORD MSG003
  
```

7109 046232  
 046232 104410  
 046234 001340

ESCAPE TST ; AND ABORT TEST

```

TRAP C$ESCAPE
.WORD L10045-
  
```

7110  
 7111 046236 004737 030264'  
 7112  
 7113 046242 103010  
 7114 046244

20\$:

```

JSR PC,CLRDNI ; WRITE ONE TO CLEAR DNI
; ERROR ?
BCC 30$ ; NO
FTL
  
```



```

046244 004737 026652'      JSR      PC,CHKFTI      ; 'FATL' BIT SET?
7115 046250      ERRHRD 125.,ERR006,MSG003 ; YES, REPORT ERROR
      046250 104456      TRAP      C$ERHRD
      046252 000175      .WORD    125
      046254 023030'      .WORD    ERR006
      046256 021736'      .WORD    MSG003
7116 046260      ESCAPE TST      ; AND ABORT TEST
      046260 104410      TRAP      C$ESCAPE
      046262 001312      .WORD    L10045-.
7117
7118 046264      ; 30$:
7119 046264 004737 030212'      JSR      PC,CLRBUF      ; CLEAR TRANSMIT AND RECV BUFFERS
7120 046270 004737 031732'      JSR      PC,LDDFLT      ; LOAD DEFAULT PHY.ADDRESS TABLES
7121 046274 004737 032032'      JSR      PC,LDPCSR      ; ADDRESS OF PCBB -> PCSR2!3
7122 046300 012777 004100 131716      MOV      @DNI!INTE,@PCSR0
7123 046306 112777 000101 131710      MOVB    @INTE!GETPCB,@PCSR0 ; ISSUE GET_PCBB PORT COMMAND
7124 046314 004737 026550'      JSR      PC,CHKDNI      ; DNI?
7125 046320 103010      BCC      40$           ; YES
7126 046322      FTL
      046322 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
7127 046326      ERRHRD 126.,ERR009,MSG003 ; NO, REPORT ERROR
      046326 104456      TRAP      C$ERHRD
      046330 000176      .WORD    126
      046332 023245'      .WORD    ERR009
      046334 021736'      .WORD    MSG003
7128 046336      ESCAPE TST      ; AND ABORT TEST
      046336 104410      TRAP      C$ESCAPE
      046340 001234      .WORD    L10045-.
7129
7130 046342 004737 030264'      ; 40$: JSR      PC,CLRDN1      ; WRITE ONE TO CLEAR DNI
7131      BCC      50$           ; ERROR ?
7132 046346 103010      BCC      50$           ; NO
7133 046350      FTL
      046350 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
7134 046354      ERRHRD 127.,ERR006,MSG003 ; YES, REPORT ERROR
      046354 104456      TRAP      C$ERHRD
      046356 000177      .WORD    127
      046360 023030'      .WORD    ERR006
      046362 021736'      .WORD    MSG003
7135 046364      ESCAPE TST      ; AND ABORT TEST
      046364 104410      TRAP      C$ESCAPE
      046366 001206      .WORD    L10045-.
7136
7137      ;WRITE MODE REGISTER = INTERNAL LOOPBACK, PROM AND DISABLE XMIT CRC MODE
7138
7139 046370 012705 012620'      ; 50$: MOV      @WTFMOD2,R5      ; WRITE MODE FUNCTION, DISABLE
7140      ; TRANSMIT CRC
7141 046374 004737 032002'      JSR      PC,LDPCBB      ; LOAD FUNCTION -> PCBB
7142 046400 013737 016464' 000302'      MOV      MODE15,PCBB+2 ; LOAD MODE REGISTER
7143 046406 012777 004100 131610      MOV      @DNI!INTE,@PCSR0
7144 046414 112777 000102 131602      MOVB    @INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
7145 046422 004737 026550'      JSR      PC,CHKDNI      ; DNI ?
    
```

```

7146 046426 103010          BCC      60$          ; YES
7147 046430          FTL
      046430 004737 026652' JSR      PC,CHKFTL   ; 'FATL' BIT SET?
7148 046434          ERRHRD   130.,ERR010,MSG003 ; NO, REPORT ERROR
      046434 104456          TRAP     C$ERHRD
      046436 000202          .WORD   130
      046440 023331'        .WORD   ERR010
      046442 021736'        .WORD   MSG003
7149 046444          ESCAPE   TST          ; AND ABORT TEST
      046444 104410          TRAP     C$ESCAPE
      046446 001126          .WORD   L10045-.
7150          ;
7151 046450 004737 030264' 60$: JSR      PC,CLRDNI   ; WRITE ONE TO CLEAR DNI
7152          ; ERROR ?
7153 046454 103010          BCC      70$          ; NO
7154 046456          FTL
      046456 004737 026652' JSR      PC,CHKFTL   ; 'FATL' BIT SET?
7155 046462          ERRHRD   131.,ERR006,MSG003 ; YES, REPORT ERROR
      046462 104456          TRAP     C$ERHRD
      046464 000203          .WORD   131
      046466 023030'        .WORD   ERR006
      046470 021736'        .WORD   MSG003
7156 046472          ESCAPE   TST          ; AND ABORT TEST
      046472 104410          TRAP     C$ESCAPE
      046474 001100          .WORD   L10045-.
7157          ;WRITE RING FORMAT
7158          ;
7159          70$:
7160 046476 012705 012540' MOV     @WTRNGS,R5   ; DEFAULT WRITE RING FORMAT FUNCTION
7161 046502 004737 032002' JSR     PC,LDPCBB    ; LOAD FUNCTION -> PCBB
7162 046506 012705 012704' MOV     @RFRMT,R5   ; DEFAULT RING FORMAT
7163 046512 012700 000006' MOV     @6,R0       ; FORMAT = SIX WORDS
7164 046516 004737 032260' JSR     PC,LDUDBB    ; LOAD RING FORMAT -> UDBB
7165 046522 012777 004100 131474 MOV     @DNI!INTE,@PCSR0
7166 046530 112777 000102 131466 MOVB    @INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
7167 046536 004737 026550' JSR     PC,CHKDNI   ; DNI ?
7168 046542 103010          BCC      80$          ; YES
7169 046544          FTL
      046544 004737 026652' JSR      PC,CHKFTL   ; 'FATL' BIT SET?
7170 046550          ERRHRD   132.,ERR010,MSG003 ; NO, REPORT ERROR
      046550 104456          TRAP     C$ERHRD
      046552 000204          .WORD   132
      046554 023331'        .WORD   ERR010
      046556 021736'        .WORD   MSG003
7171 046560          ESCAPE   TST          ; AND ABORT TEST
      046560 104410          TRAP     C$ESCAPE
      046562 001012          .WORD   L10045-.
7172          ;
7173 046564 004737 030264' 80$: JSR      PC,CLRDNI   ; WRITE ONE TO CLEAR DNI
7174          ; ERROR ?
7175 046570 103010          BCC      90$          ; NO
    
```

```

7176 046572          FTL
      046572 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
7177 046576          ERRHRD 133.,ERR006,MSG003      ; YES, REPORT ERROR
      046576 104456          TRAP      C$ERHRD
      046600 000205          .WORD    133
      046602 023030'        .WORD    ERR006
      046604 021736'        .WORD    MSG003
7178 046606          ESCAPE TST          ; AND ABORT TEST
      046606 104410          TRAP      C$ESCAPE
      046610 000764          .WORD    L10045-.
7179
7180          ;WRITE PHYSICAL ADDRESS
7181
7182 046612          90$:
7183 046612 012705 000272'      MOV      #DEFAULT,R5          ; GET DEFAULT PHYSICAL ADDRESS
7184 046616 004737 032050'      JSR      PC,LDPHYA          ; SAVE IT IN DEFAULT TABLE
7185 046622 012705 012500'      MOV      #WTPHYA,R5          ; DEFAULT WRITE PHYSICAL ADDR FUNC
7186 046626 004737 032002'      JSR      PC,LPCBB          ; LOAD FUNCTION -> PCBB
7187 046632 012777 004100 131364  MOV      #DNI!INTE,@PCSR0      ; ISSUE GET_CMD PORT COMMAND
7188 046640 112777 000102 131356  MOVB    #INTE!GETCMD,@PCSR0
7189 046646 004737 026550'      JSR      PC,CHKDNI          ; DNI ?
7190 046652 103010          BCC      100$              ; YES
7191 046654          FTL
      046654 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
7192 046660          ERRHRD 134.,ERR010,MSG003      ; NO, REPORT ERROR
      046660 104456          TRAP      C$ERHRD
      046662 000206          .WORD    134
      046664 023331'        .WORD    ERR010
      046666 021736'        .WORD    MSG003
7193 046670          ESCAPE TST          ; AND ABORT TEST
      046670 104410          TRAP      C$ESCAPE
      046672 000702          .WORD    L10045-.
7194
7195 046674 004737 030264'      ;
7196          ;100$: JSR      PC,CLRDNI          ; WRITE ONE TO CLEAR DNI
7197 046700 103010          BCC      110$              ; ERROR ?
7198 046702          FTL          ; NO
      046702 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
7199 046706          ERRHRD 135.,ERR006,MSG003      ; YES, REPORT ERROR
      046706 104456          TRAP      C$ERHRD
      046710 000207          .WORD    135
      046712 023030'        .WORD    ERR006
      046714 021736'        .WORD    MSG003
7200 046716          ESCAPE TST          ; AND ABORT TEST
      046716 104410          TRAP      C$ESCAPE
      046720 000654          .WORD    L10045 .
7201
7202          ;SET UP RINGS FOR ONE BUFFER LOOPBACK
7203
7204 046722 012705 014450'      110$: MOV      #TDRB1B,R5          ; DEFAULT ONE BUFFER TRANSMIT RING
7205 046726 004737 032164'      JSR      PC,LDTDRB          ; LOAD TDRB
  
```



7239	047116	021736'		ESCAPE	TST	; AND ABORT TEST	.WORD	MSG003
	047120						TRAP	C\$ESCAPE
	047120	104410					.WORD	L10045-.
	047122	000452						
7240								
7241	047124	004737	030500'	i	140\$: JSR	PC,CLRXTI		; WRITE ONE TO CLEAR TXI
7242								; ERROR ?
7243	047130	103010			BCC	150\$		; NO
7244	047132				FTL			
	047132	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
7245	047136				ERRHRD	141.,ERR014,MSG003		; YES, REPORT ERROR
	047136	104456					TRAP	C\$ERHRD
	047140	000215					.WORD	141
	047142	023561'					.WORD	ERR014
	047144	021736'					.WORD	MSG003
7246	047146				ESCAPE	TST		; AND ABORT TEST
	047146	104410					TRAP	C\$ESCAPE
	047150	000424					.WORD	L10045-.
7247								
7248	047152	012705	000620'	i	150\$: MOV	#TDRB,R5		; CHECK TDRB OWNERSHIP
7249	047156	004737	027024'		JSR	PC,CHKOWN		; OWN = PORT DRIVER ?
7250	047162	103010			BCC	160\$		; YES
7251	047164				FTL			
	047164	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
7252	047170				ERRHRD	142.,ERR018		; NO, REPORT ERROR
	047170	104456					TRAP	C\$ERHRD
	047172	000216					.WORD	142
	047174	024026'					.WORD	ERR018
	047176	000000					.WORD	0
7253	047200				ESCAPE	TST		; AND ABORT TEST
	047200	104410					TRAP	C\$ESCAPE
	047202	000372					.WORD	L10045-.
7254								
7255	047204	012705	016274'	i	160\$: MOV	#TDR15A,R5		; POINT TO EXPECTED TDRB
7256	047210	004737	032370'		JSR	PC,LXXTDR		; LOAD INTO XTDRBO TABLE
7257	047214	012705	000620'		MOV	#TDRB,R5		; CHECK TDRB
7258	047220	004737	027500'		JSR	PC,CHKTDR		; ERRORS ?
7259	047224	103010			BCC	170\$		; NO
7260	047226				FTL			
	047226	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
7261	047232				ERRHRD	143.,ERR020,MSG005		; YES, REPORT ERROR
	047232	104456					TRAP	C\$ERHRD
	047234	000217					.WORD	143
	047236	024206'					.WORD	ERR020
	047240	022042'					.WORD	MSG005
7262	047242				ESCAPE	TST		; AND ABORT TEST
	047242	104410					TRAP	C\$ESCAPE
	047244	000330					.WORD	L10045-.
7263								
7264	047246	004737	027316'	i	170\$: JSR	PC,CHKRXI		; RXI ?
7265	047252	103010			BCC	180\$		; YES

```

7266 047254          FTL
      047254 004737 026652'      JSR    PC,CHKFTL          ; 'FATL' BIT SET?
7267 047260          ERRHRD 144.,ERR015,MSG003      ; NO, REPORT ERROR
      047260 104456          TRAP  C#ERHRD
      047262 000220          .WORD 144
      047264 023627'        .WORD  ERR015
      047266 021736'        .WORD  MSG003
7268 047270          ESCAPE TST          ; AND ABORT TEST
      047270 104410          TRAP  C#ESCAPE
      047272 000302          .WORD  L10045-.
7269 047274 004737 030432'      ;
180$: JSR    PC,CLRRXI          ; WRITE ONE TO CLEAR RXI
7270 047274 004737 030432'      ; ERROR ?
7271 047274 004737 030432'      ; NO
7272 047300 103010          BCC    190$
7273 047302          FTL
      047302 004737 026652'      JSR    PC,CHKFTL          ; 'FATL' BIT SET?
7274 047306          ERRHRD 145.,:R016,MSG003      ; YES, REPORT ERROR
      047306 104456          TRAP  C#ERHRD
      047310 000221          .WORD 145
      047312 023660'        .WORD  ERR016
      047314 021736'        .WORD  MSG003
7275 047316          ESCAPE TST          ; AND ABORT TEST
      047316 104410          TRAP  C#ESCAPE
      047320 000254          .WORD  L10045-.
7276 047322 012705 000660'      ;
190$: MOV    #RDRB,R5          ; CHECK RDRB OWNERSHIP
7277 047322 012705 000660'      JSR    PC,CHKOWN         ; OWN = PORT DRIVER ?
7278 047326 004737 027024'      BCC    200$
7279 047332 103010          FTL
7280 047334          FTL
      047334 004737 026652'      JSR    PC,CHKFTL          ; 'FATL' BIT SET?
7281 047340          ERRHRD 146.,ERR017      ; NO, REPORT ERROR
      047340 104456          TRAP  C#ERHRD
      047342 000222          .WORD 146
      047344 023726'        .WORD  ERR017
      047346 000000          .WORD  0
7282 047350          ESCAPE TST          ; AND ABORT TEST
      047350 104410          TRAP  C#ESCAPE
      047352 000222          .WORD  L10045-.
7283 047354 012705 016404'      ;
200$: MOV    #RDR15A,R5       ; POINT TO EXPECTED RDRB
7284 047354 012705 016404'      JSR    PC,LDXRDR        ; LOAD INTO XRDRBO TABLE
7285 047360 004737 032340'      MOV    #RDRB,R5        ; CHECK RDRB
7286 047364 012705 000660'      JSR    PC,CHKRDR       ; ERRORS ?
7287 047370 004737 027206'      BCC    210$
7288 047374 103010          FTL
7289 047376          FTL
      047376 004737 026652'      JSR    PC,CHKFTL          ; 'FATL' BIT SET?
7290 047402          ERRHRD 147.,ERR021,MSG006      ; YES, REPORT ERROR
      047402 104456          TRAP  C#ERHRD
      047404 000223          .WORD 147
  
```

```

047406 024267'
047410 022204'
7291 047412          ESCAPE TST          ; AND ABORT TEST          .WORD  ERR021
047412 104410          ;                               .WORD  MSG006
047414 000160          ;                               TRAP   C$ESCAPE
7292          ;                               .WORD  L10045-.
7293          ;COMPARE RBUF WITH TBUF
7294
7295 047416 013705 016560' 210$:  MOV    BYTCNT,R5          ; COMPARE DATA
7296 047422 004737 030712'  JSR    PC,CMPDAT          ; DATA COMPARE ERROR ?
7297 047426 103006          BCC    230$              ; NO
7298 047430          ERRHRD 150.,ERR022,MSG007 ; YES, REPORT ERROR
047430 104456          TRAP   C$ERHRD
047432 000226          .WORD  150
047434 024350'          .WORD  ERR022
047436 022346'          .WORD  MSG007
7299 047440          ESCAPE TST          ; AND ABORT TEST          TRAP   C$ESCAPE
047440 104410          ;                               .WORD  L10045-.
047442 000132
7300
7301 047444 012777 004100 130552 ; 230$: MOV    #DNI!INTE,@PCSR0
7302 047452 112777 000117 130544 ; MOVB  #INTE!STOP,@PCSR0 ; ISSUE STOP PORT COMMAND
7303 047460 004737 026550' ; JSR    PC,CHKDNI          ; DNI ?
7304 047464 103010          BCC    240$              ; YES
7305 047466          FTL
047466 004737 026652' ; JSR    PC,CHKFTL          ; 'FATL' BIT SET?
7306 047472          ERRHRD 151.,ERR019,MSG003 ; NO, REPORT ERROR
047472 104456          TRAP   C$ERHRD
047474 000227          .WORD  151
047476 024126'          .WORD  ERR019
047500 021736'          .WORD  MSG003
7307 047502          ESCAPE TST          ; AND ABORT TEST          TRAP   C$ESCAPE
047502 104410          ;                               .WORD  L10045-.
047504 000070
7308
7309 047506 004737 030264' ; 240$: JSR    PC,CLRDN1          ; WRITE ONE TO CLEAR DNI
7310          ;                               ; ERROR ?
7311 047512 103010          BCC    250$              ; NO
7312 047514          FTL
047514 004737 026652' ; JSR    PC,CHKFTL          ; 'FATL' BIT SET?
7313 047520          ERRHRD 161.,ERR006,MSG003 ; YES, REPORT ERROR
047520 104456          TRAP   C$ERHRD
047522 000241          .WORD  161
047524 023030'          .WORD  ERR006
047526 021736'          .WORD  MSG003
7314 047530          ESCAPE TST          ; AND ABORT TEST          TRAP   C$ESCAPE
047530 104410          ;                               .WORD  L10045-.
047532 000042
7315 047534          250$: JSR    PC,CLINTR          ; INSURE DELUA INTR BITS CLEAR
7316 047534 004737 030346'
7317
7318 047540          EXIT   TST
047540 104432          TRAP   C$EXIT
    
```

.WORD L10045 .

047542 000032

7319

7320

7321

7322

047544	104	105	114
047547	125	101	040
047552	106	117	122
047555	103	105	040
047560	103	122	103
047563	040	105	122
047566	122	117	122
047571	040	000	

;LOCAL TEST MESSAGE

T15ID: .ASCIZ 'DELUA FORCE CRC ERROR '

7323

7324

7325

047574	
047574	
047574	104401

.EVEN

ENDTST

L10045:

TRAP C\$ETST



7327  
 7328  
 7329  
 7330  
 7331  
 7332  
 7333  
 7334  
 7335  
 7336  
 7337  
 7338  
 7339  
 7340  
 7341  
 7342  
 7343  
 7344  
 7345  
 7346  
 7347  
 7348  
 7349  
 7350  
 7351  
 7352  
 7353  
 7354  
 7355  
 7356  
 7357  
 7358  
 7359  
 7360  
 7361  
 7362  
 7363  
 7364  
 7365  
 7366  
 7367

.SBTTL TEST 16: NO RECEIVE BUFFER TEST

```

*****
:
: THIS TEST VERIFIES THAT A RCBI ERROR CAN BE DETECTED.
: THIS ERROR WILL ONLY OCCUR AFTER 47. RECEIVE ENTRIES
: (SIZE OF INTERNAL RECEIVE BUFFER), FOR CASE WHERE NO
: RECEIVE BUFFERS ARE OWNED BY THE DELUA.
:
: TEST SEQUENCE:
: 1. WRITE MODE REGISTER = INTERNAL LOOPBACK AND PROM MODE
: 2. WRITE RING FORMAT
: 3. WRITE PHYSICAL ADDRESS
: 4. SET UP RINGS AND BUFFERS WITH 49.
:   TRANSMIT PACKETS, AND NO RECEIVE
:   BUFFERS OWNED BY THE DELUA.
: 5. INSURE 'RECEIVE PACKET LOST' COUNTER IS CLEAR
: 6. ISSUE START
: 7. AFTER EACH OF THE FIRST 47. TRANSMISSION'S,
:   'RECEIVE PACKET LOST' COUNTER SHOULD BE CLEAR,
:   AND THERE SHOULD BE NO 'RCBI' ERROR
: 8. FOLLOWING NEXT (48TH) TRANSMISSION, 'RCBI' ERROR
:   BIT IN PCSRO SHOULD SET AND, 'RECEIVE PACKET LOST'
:   COUNTER SHOULD BE INCREMENTED TO A ONE.
: 9. ISSUE STOP
:
*****
  
```

BGNTST

T16::

PNTMAC T16ID

047576 012704 050650'  
 047602 004737 032734'

```

MOV @T16ID,R4 ;GET POINTER TO TEST NAME MESSAGE
JSR PC,PNTID ;PRINT TEST NUMBER AND NAME
  
```

END OF MACRO EXPANSION OF 'PNTMAC'

047606 004737 033434'  
 047612 103034  
 047614 012777 004100 130402  
 047622 112777 000140 130374  
 047630 004737 027736'  
 047634 103010  
 047636

```

JSR PC,TINIT ; IS A DEVICE RESET NEEDED?
BCC 30$ ; NO
MOV @DNI!INTE,@PCSRO ; ENABLE INTERRUPTS
MOVB @INTE!RSET,@PCSRO ; YES, RESET DELUA
JSR PC,CKDNI ; DNI ?
BCC 20$ ; YES
FTL
  
```

047636 004737 026652'

```

JSR PC,CHKFTL ; 'FATL' BIT SET?
  
```

047642 104456  
 047644 000242  
 047646 023030'  
 047650 021736'  
 047652 104410  
 047654 001026

```

ERRHRD 162.,ERR006,MSG003 ; NO. REPORT ERROR
  
```

```

TRAP C$ERRHRD
.WORD 162
.WORD ERR006
.WORD MSG003
  
```

```

ESCAPE TST ; AND ABORT TEST
  
```

```

TRAP C$ESCAPE
.WORD L10046
  
```

```

7368 047656 004737 030264'      20$: JSR    PC,CLRDN1      ; WRITE ONE TO CLEAR DNI
7369                                     ; ERROR ?
7370 047662 103010                BCC    30$              ; NO
7371 047664                                FTL

      047664 004737 026652'      JSR    PC,CHKFTL        ; 'FATL' BIT SET?
7372 047670                ERRHRD 163.,ERR006,MSG003 ; YES, REPORT ERROR
      047670 104456                                TRAP   C$ERHRD
      047672 000243                                .WORD 163
      047674 023030'                                .WORD ERR006
      047676 021736'                                .WORD MSG003
7373 047700                ESCAPE TST          ; AND ABORT TEST
      047700 104410                                TRAP   C$ESCAPE
      047702 001000                                .WORD L!0046-.
7374
7375 047704                ;
      30$: JSR    PC,CLRBUF      ; CLEAR XMIT, RECV BUFFERS
7376 047704 004737 030212'      JSR    PC,LDDFLT        ; LOAD DEFAULT PHY.ADDRESS TABLES
7377 047710 004737 031732'      JSR    PC,LDPCSR        ; ADDRESS OF PCBB -> PCSR2!3
7378 047714 004737 032032'      JSR    PC,LDPCSR        ;
7379 047720 012777 004100 130276 MOV    %DNI!INTE,%PCSR0 ; ENABLE INTERRUPTS
7380 047726 112777 000101 130270 MOVB   %INTE!GETPCB,%PCSR0 ; ISSUE GET_PCBB PORT COMMAND
7381 047734 004737 026550'      JSR    PC,CHKDNI        ; DNI?
7382 047740 103010                BCC    40$              ; YES
7383 047742                                FTL

      047742 004737 026652'      JSR    PC,CHKFTL        ; 'FATL' BIT SET?
7384 047746                ERRHRD 164.,ERR009,MSG003 ; NO, REPORT ERROR
      047746 104456                                TRAP   C$ERHRD
      047750 000244                                .WORD 164
      047752 023245'                                .WORD ERR009
      047754 021736'                                .WORD MSG003
7385 047756                ESCAPE TST          ; AND ABORT TEST
      047756 104410                                TRAP   C$ESCAPE
      047760 000722                                .WORD L10046 .
7386
7387 047762 004737 030264'      40$: JSR    PC,CLRDN1      ; WRITE ONE TO CLEAR DNI
7388                                     ; ERROR ?
7389 047766 103010                BCC    50$              ; NO
7390 047770                                FTL

      047770 004737 026652'      JSR    PC,CHKFTL        ; 'FATL' BIT SET?
7391 047774                ERRHRD 165.,ERR006,MSG003 ; YES, REPORT ERROR
      047774 104456                                TRAP   C$ERHRD
      047776 000245                                .WORD 165
      050000 023030'                                .WORD ERR006
      050002 021736'                                .WORD MSG003
7392 050004                ESCAPE TST          ; AND ABORT TEST
      050004 104410                                TRAP   C$ESCAPE
      050006 000674                                .WORD L10046 .
7393
7394                ;WRITE MODE REGISTER = INTERNAL LOOPBACK AND PROM MODE
7395
7396 050010 012705 012600'      50$: MOV    %WTMODE,R5      ; DEFAULT WRITE MODE FUNCTION
7397 050014 004737 032002'      JSR    PC,LDPCCB        ; LOAD FUNCTION -> PCBB
  
```

```

7398 050020 012777 004100 130176      MOV    #DNI!INTE,@PCSR0      ; ENABLE INTERRUPTS
7399 050026 112777 000102 130170      MOVB  #INTE!GETCMD,@PCSR0   ; ISSUE GET_CMD PORT COMMAND
7400 050034 004737 026550'          JSR   PC,CHKDNI              ; DNI ?
7401 050040 103010                    BCC   60$                   ; YES
7402 050042                    FTL

      050042 004737 026652'          JSR   PC,CHKFTL              ; 'FATL' BIT SET?

7403 050046                    ERRHRD 166.,ERR010,MSG003     ; NO, REPORT ERROR
      050046 104456                    TRAP  C$ERHRD
      050050 000246                    .WORD 166
      050052 023331'                   .WORD ERR010
      050054 021736'                   .WORD MSG003

7404 050056                    ESCAPE TST                   ; AND ABORT TEST
      050056 104410                    TRAP  C$ESCAPE
      050060 000622                    .WORD L10046-.

7405
7406 050062 004737 030264'          ; 60$: JSR   PC,CLRDN1       ; WRITE ONE TO CLEAR DNI
7407
7408 050066 103010                    BCC   70$                   ; ERROR ?
7409 050070                    FTL                          ; NO

      050070 004737 026652'          JSR   PC,CHKFTL              ; 'FATL' BIT SET?

7410 050074                    ERRHRD 167.,ERR006,MSG003     ; YES, REPORT ERROR
      050074 104456                    TRAP  C$ERHRD
      050076 000247                    .WORD 167
      050100 023030'                   .WORD ERR006
      050102 021736'                   .WORD MSG003

7411 050104                    ESCAPE TST                   ; AND ABOPT TEST
      050104 104410                    TRAP  C$ESCAPE
      050106 000574                    .WORD L10046-.

7412
7413 ;WRITE RING FORMAT (41 TRANSMIT ENTRIES)
7414
7415 050110 012705 012540'          70$: MOV    #WTRNGS,R5       ; DEFAULT WRITE RING FORMAT FUNCTION
7416 050114 004737 032002'          JSR   PC,LDPCCB              ; LOAD FUNCTION -> PCBB
7417 050114 012705 012720'          MOV    #FRMTX,R5            ; DEFAL" T RING FORMAT
7418 050124 012700 000006          MOV    #6,R0                 ; FORMAI = SIX WORDS
7419 050130 004737 032260'          JSR   PC,LDUDBB              ; LOAD RING FORMAT -> UDBB
7420 050134 012777 004100 130062      MOV    #DNI!INTE,@PCSR0     ; ENABLE INTERRUPTS
7421 050142 112777 000102 130054      MOVB  #INTE!GETCMD,@PCSR0   ; ISSUE GET_CMD PORT COMMAND
7422 050150 004737 026550'          JSR   PC,CHKDNI              ; DNI ?
7423 050154 103010                    BCC   80$                   ; YES
7424 050156                    FTL

      050156 004737 026652'          JSR   PC,CHKFTL              ; 'FATL' BIT SET?

7425 050162                    ERRHRD 170.,ERR010,MSG003     ; NO, REPORT ERROR
      050162 104456                    TRAP  C$ERHRD
      050164 000252                    .WORD 170
      050166 023331'                   .WORD ERR010
      050170 021736'                   .WORD MSG003

7426 050172                    ESCAPE TST                   ; AND ABORT TEST
      050172 104410                    TRAP  C$ESCAPE
      050174 000506                    .WORD L10046-.

7427 ;
    
```

```

7428 050176 004737 030264'      80$: JSR    PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
7429                                ; ERROR ?
7430 050202 103010              BCC    90$              ; NO
7431 050204                                FTL
                                JSR    PC,CHKFTL      ; 'FATL' BIT SET?
                                ERRHRD 171.,ERR006,MSG003 ; YES, REPORT ERROR
7432 050210                                TRAP  C$ERHRD
050210 104456                                .WORD 171
050212 000253                                .WORD ERR006
050214 023030'                                .WORD MSG003
050216 021736'
7433 050220                                ESCAPE TST              ; AND ABORT TEST
050220 104410                                TRAP  C$ESCAPE
050222 000460                                .WORD L10046-.

7434                                ;
7435                                ;WRITE PHYSICAL ADDRESS
7436                                ;
7437 050224      90$:
7438 050224 012705 000272'      MOV    #DEFAULT,R5      ; GET DEFAULT PHYSICAL ADDRESS
7439 050230 004737 032050'      JSR    PC,LDPHYA        ; SAVE IN DEFAULT TABLE
7440 050234 012705 012500'      MOV    #WTPHYA,R5      ; DEFAULT WRITE PHYSICAL ADDR FUNC
7441 050240 004737 032002'      JSR    PC,LDPCCB        ; LOAD FUNCTION -> PCBB
7442 050244 012777 004100 127752  MOV    #DNI!INTE,@PCSRO ; ENABLE INTERRUPTS
7443 050252 112777 000102 127744  MOVB  #INTE!GETCMD,@PCSRO ; ISSUE GET_CMD PORT COMMAND
7444 050260 004737 026550'      JSR    PC,CHKDNI        ; DNI ?
7445 050264 103010              BCC    100$             ; YES
7446 050266                                FTL
                                JSR    PC,CHKFTL      ; 'FATL' BIT SET?
                                ERRHRD 172.,ERR010,MSG003 ; NO, REPORT ERROR
7447 050272                                TRAP  C$ERHRD
050272 104456                                .WORD 172
050274 000254                                .WORD ERR010
050276 023331'                                .WORD MSG003
050300 021736'
7448 050302                                ESCAPE TST              ; AND ABORT TEST
050302 104410                                TRAP  C$ESCAPE
050304 000376                                .WORD L10046-.

7449                                ;
7450 050306 004737 030264'      100$: JSR    PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
7451                                ; ERROR ?
7452 050312 103010              BCC    110$             ; NO
7453 050314                                FTL
                                JSR    PC,CHKFTL      ; 'FATL' BIT SET?
                                ERRHRD 173.,ERR006,MSG003 ; YES, REPORT ERROR
7454 050320                                TRAP  C$ERHRD
050320 104456                                .WORD 173
050322 000255                                .WORD ERR006
050324 023030'                                .WORD MSG003
050326 021736'
7455 050330                                ESCAPE TST              ; AND ABORT TEST
050330 104410                                TRAP  C$ESCAPE
050332 000350                                .WORD L10046-.

7456                                ;
7457                                ;SET UP RINGS FOR 50. TRANSMIT PACKETS

```

```

7458 ;AND NO RECEIVE BUFFERS OWNED BY DELUA
7459
7460 050334 012705 014740' 110$: MOV #TDRBXX,R5 ; TRANSMIT RING
7461 050340 004737 032222' JSR PC,LDRDX ; LOAD TDRBX
7462 050344 012705 013010' MOV #RDRB1B,R5 ; DEFAULT RECEIVE RING (NO BUFFERS)
7463 050350 004737 032126' JSR PC,LDRDRX ; LOAD RDRX
7464
7465 ;SET UP BUFFERS AND START
7466
7467 050354 005037 016562' CLR DDCRC ; NO CRC
7468 050360 012737 000006 016560' MOV #6,BYTCNT ; BYTES/PACKET
7469 050366 004737 033006' JSR PC,SETBUF ; SET UP BUFFERS
7470 050372 012777 004100 127624 MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
7471 050400 112777 000104 127616 MOVB #INTE!START,@PCSR0 ; ISSUE START PORT COMMAND
7472 050406 004737 026550' JSR PC,CHKDNI ; DNI?
7473 050412 103010 BCC 130$ ; YES
7474 050414 FTL
;
050414 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?
;
7475 050420 ERRHRD 174.,ERR012,MSG003 ; NO, REPORT ERROR
;
050420 104456 TRAP C$ERHRD
050422 000256 .WORD 174
050424 023447' .WORD ERR012
050426 021736' .WORD MSG003
;
7476 050430 ESCAPE TST ; AND ABORT TEST
;
050430 104410 TRAP C$ESCAPE
050432 000250 .WORD L10046-.
;
7477 ;
7478 ;Don't use subroutine CLRDNI until check PCSRO upper byte status
7479 ;
7480 ;WAIT FOR RCBI SET IN PCSRO
7481 ;
7482 050434 130$:
7483 050434 004737 027102' JSR PC,CHKRCE ; BUFFER AVAIL ERROR?
7484 050440 103016 BCC 180$ ; YES, SKIP ERROR PRINTOUT
7485 050442 013737 000236' 016514' MOV PCSROC,EPCSR0 ; SET UP DATA FOR
7486 050450 017737 127552 016516' MOV @PCSR1,EPCSR1 ; ERROR PRINTOUT
7487 050456 FTL
;
050456 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?
;
7488 050462 ERRHRD 175.,ERR025,MSG003 ; NO, REPORT ERROR
;
050462 104456 TRAP C$ERHRD
050464 000257 .WORD 175
050466 024506' .WORD ERR025
050470 021736' .WORD MSG003
;
7489 050472 ESCAPE TST ; AND ABORT TEST
;
050472 104410 TRAP C$ESCAPE
050474 000206 .WORD L10046 .
;
7490 ;
7491 ;Now can clear PCSRO upper byte
7492 ;
7493 050476 004737 030346' 180$: JSR PC,CLINTR ; WRITE ONE'S TO CLEAR UPPER BYTE
7494 050502 017700 127516 MOV @PCSR0,RO ; READ UPPER BYTE
7495 050506 032700 175400 BIT #CLINTB,RO ; ANY INTERRUPT BITS STILL SET?
7496 050512 001416 BEQ 230$ ; IF NO ERROR, SKIP ERROR REPORT
    
```

```

7497 050514 017737 127504 016514'      MOV    @PCSR0,EPCSR0      ; SAVE PCSRO AND
7498 050522 017737 127500 016516'      MOV    @PCSR1,EPCSR1      ;   PCSR1 FOR ERROR REPORT
7499 050530                                FTL

      050530 004737 026652'              JSR    PC,CHKFTL          ; 'FATL' BIT SET?
7500 050534                                ERRHRD 176.,ERR045,MSG003 ; YES, REPORT ERROR
      050534 104456                                TRAP   C$ERHRD
      050536 000260                                .WORD 176
      050540 026274'                                .WORD ERR045
      050542 021736'                                .WORD MSG003
7501 050544                                ESCAPE TST                ; AND ABORT TEST
      050544 104410                                TRAP   C$ESCAPE
      050546 000134                                .WORD L10046-.

7502
7503
7504 050550                                ;
7505 050550 012777 004100 127446      ; 230$: MOV    @DNI!INTE,@PCSR0      ; ENABLE INTERRUPTS
7506 050556 112777 000117 127440      ; MOVB  @INTE!STOP,@PCSR0    ; ISSUE STOP PORT COMMAND
7507 050564 004737 026550'              ; JSR    PC,CHKDNI          ; DNI ?
7508 050570 103010                        ; BCC    240$               ; YES
7509 050572                                ; FTL

      050572 004737 026652'              JSR    PC,CHKFTL          ; 'FATL' BIT SET?
7510 050576                                ERRHRD 200.,ERR019,MSG003 ; NO, REPORT ERROR
      050576 104456                                TRAP   C$ERHRD
      050600 000310                                .WORD 200
      050602 024126'                                .WORD ERR019
      050604 021736'                                .WORD MSG003
7511 050606                                ESCAPE TST                ; AND ABORT TEST
      050606 104410                                TRAP   C$ESCAPE
      050610 000072                                .WORD L10046 .

7512
7513 050612 004737 030264'              ; 240$: JSR    PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
7514                                ; BCC    250$               ; ERROR ?
7515 050616 103010                        ; FTL                        ; NO
7516 050620                                ;

      050620 004737 026652'              JSR    PC,CHKFTL          ; 'FATL' BIT SET?
7517 050624                                ERRHRD 201.,ERR006,MSG003 ; YES, REPORT ERROR
      050624 104456                                TRAP   C$ERHRD
      050626 000311                                .WORD 201
      050630 023030'                                .WORD ERR006
      050632 021736'                                .WORD MSG003
7518 050634                                ESCAPE TST                ; AND ABORT TEST
      050634 104410                                TRAP   C$ESCAPE
      050636 000044                                .WORD L10046-.

7519 050640                                ; 250$:
7520 050640 004737 030346'              JSR    PC,CLINTR         ; INSURE DELUA INTR BITS CLEAR
7521                                ;
7522 050644                                EXIT    TST
      050644 104432                                TRAP   C$EXIT
      050646 000034                                .WORD L10046 .

7523
7524                                ;LOCAL TEST MESSAGE

```

7525  
7526 050650 104 105 114 T16ID: .ASCIZ 'DELUA NO RECEIVE BUFFER '  
050653 125 101 040  
050656 116 117 040  
050661 122 105 103  
050664 105 111 126  
050667 105 040 102  
050672 125 106 106  
050675 105 122 040  
050700 000

7527  
7528 .EVEN  
7529  
7530 050702 ENDTST  
050702  
050702 104401

L10046: TRAP C#ETST

7532  
 7533  
 7534  
 7535  
 7536  
 7537  
 7538  
 7539  
 7540  
 7541  
 7542  
 7543  
 7544  
 7545  
 7546  
 7547  
 7548  
 7549  
 7550

.SBTTL TEST 17: DISABLE RECEIVE CHAINING TEST

```

*****
:
: THIS TEST VERIFIES DISABLE DATA CHAINING MODE.
: AN INTERNAL LOOPBACK IS PERFORMED WITH RECEIVE BUFFERS CHAINED
: WHILE IN DISABLE DATA CHAINING MODE.
: A NCHN ERROR IS EXPECTED IN THE RECEIVE DESCRIPTOR RING.
:
: TEST SEQUENCE:
:   1. WRITE MODE REGISTER = INTERNAL LOOPBACK, PROM,
:     and DISABLE DATA CHAINING MODE
:   2. WRITE RING FORMAT
:   3. WRITE PHYSICAL ADDRESS
:   4. SET UP RINGS AND BUFFERS FOR RECEIVE DATA CHAINING
:   5. ISSUE START
:   6. CHECK FOR NCHN ERROR IN RDRB+6
:   7. ISSUE STOP
:
*****
  
```

7553 050704  
 050704  
 7554  
 7555 050704

BGNTST

T17::

PNTMAC T17ID

050704 012704 052260'  
 050710 004737 032734'

```

MOV #T17ID,R4 ;GET POINTER TO TEST NAME MESSAGE
JSR PC,PNTID ;PRINT TEST NUMBER AND NAME
  
```

: END OF MACRO EXPANSION OF 'PNTMAC'

7556 050714 004737 033434'  
 7557 050720 103034  
 7558 050722 012777 004100 127274  
 7559 050730 112777 000140 127266  
 7560 050736 004737 027736'  
 7561 050742 103010  
 7562 050744

```

JSR PC,TINIT ; IS A DEVICE RESET NEEDED?
BCC 30$ ; NO
MOV #DNI!INTE,@PCSR0 ; PRECONDITION INTR EN.
MOV #INTE!RSET,@PCSR0 ; YES, RESET DELUA
JSR PC,CKDNI ; DNI ?
BCC 20$ ; YES
FTL
  
```

050744 004737 026652'

JSR PC,CHKFTL ; 'FATL' BIT SET?

7563 050750  
 050750 104456  
 050752 000312  
 050754 023030'  
 050756 021736'

ERRHRD 202.,ERR006,MSG003 ; NO. REPORT ERROR

```

TRAP C$ERHRD
.WORD 202
.WORD ERR006
.WORD MSG003
  
```

7564 050760  
 050760 104410  
 050762 001336

ESCAPE TST ; AND ABORT TEST

```

TRAP C$ESCAPE
.WORD L10047-.
  
```

7565  
 7566 050764 004737 030264'  
 7567  
 7568 050770 103010  
 7569 050772

```

; 20$: JSR PC,CLRDNI ; WRITE ONE TO CLEAR DNI
; ERROR ?
; NO
  
```

050772 004737 026652'

JSR PC,CHKFTL ; 'FATL' BIT SET?



```

7570 050776          ERRHRD 203.,ERR006,MSG003      ; YES, REPORT ERROR
      050776 104456          TRAP          C$ERHRD
      051000 000313          .WORD        203
      051002 023030'        .WORD        ERR006
      051004 021736'        .WORD        MSG003
7571 051006          ESCAPE TST                    ; AND ABORT TEST
      051006 104410          TRAP          C$ESCAPE
      051010 001310          .WORD        L10047-.
7572
7573 051012          ;
7574 051012 004737 030212' 30$: JSR      PC,CLRBUF          ; CLEAR XMIT,RECV BUFFERS
7575 051016 004737 031732' JSR      PC,LDDFLT         ; LOAD DEFAULT PHY. ADDRESS TABLES
7576 051022 004737 032032' JSR      PC,LDPCSR        ; ADDRESS OF PCBB -> PCSR2:3
7577 051026 012777 004100 127170 MOV     #DNI!INTE,@PCSR0
7578 051034 112777 000101 127162 MOVB   #INTE!GETPCB,@PCSR0 ; ISSUE GET_PCBB PORT COMMAND
7579 051042 004737 026550' JSR      PC,CHKDNI        ; DNI?
7580 051046 103010          BCC     40$           ; YES
7581 051050          FTL
      051050 004737 026652' JSR      PC,CHKFTL         ; 'FATL' BIT SET?
7582 051054          ERRHRD 204.,ERR009,MSG003      ; NO, REPORT ERROR
      051054 104456          TRAP          C$ERHRD
      051056 000314          .WORD        204
      051060 023245'        .WORD        ERR009
      051062 021736'        .WORD        MSG003
7583 051064          ESCAPE TST                    ; AND ABORT TEST
      051064 104410          TRAP          C$ESCAPE
      051066 001232          .WORD        L10047-.
7584
7585 051070 004737 030264' 40$: JSR      PC,CLRDNI        ; WRITE ONE TO CLEAR DNI
7586
7587 051074 103010          BCC     50$           ; ERROR ?
7588 051076          FTL                          ; NO
      051076 004737 026652' JSR      PC,CHKFTL         ; 'FATL' BIT SET?
7589 051102          ERRHRD 205.,ERR006,MSG003      ; YES, REPORT ERROR
      051102 104456          TRAP          C$ERHRD
      051104 000315          .WORD        205
      051106 023030'        .WORD        ERR006
      051110 021736'        .WORD        MSG003
7590 051112          ESCAPE TST                    ; AND ABORT TEST
      051112 104410          TRAP          C$ESCAPE
      051114 001204          .WORD        L10047-.
7591
7592          ;WRITE MODE REGISTER = INTERNAL LOOPBACK, PROM AND
7593          ;
7594          ;
7595 051116 012705 012600' 50$: MOV     #WTMODE,R5          ; DEFAULT WRITE MODE FUNCTION
7596 051122 004737 032002' JSR      PC,LDPCBB        ; LOAD FUNCTION -> PCBB
7597 051126 013737 016466' 000302' MOV     MODE17,PCBB+2    ; LOAD MODE REGISTER
7598 051134 012777 004100 127062 MOV     #DNI!INTE,@PCSR0
7599 051142 112777 000102 127054 MOVB   #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
7600 051150 004737 026550' JSR      PC,CHKDNI        ; DNI ?
7601 051154 103010          BCC     60$           ; YES
7602 051156          FTL
    
```

```

051156 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
7603 051162                ERRHRD  206.,ERR010,MSG003 ; NO, REPORT ERROR
      051162 104456                TRAP   C$ERRHRD
      051164 000316                .WORD  206
      051166 023331'                .WORD  ERR010
      051170 021736'                .WORD  MSG003
7604 051172                ESCAPE  TST              ; AND ABORT TEST
      051172 104410                TRAP   C$ESCAPE
      051174 001124                .WORD  L10047-.
7605                ;
7606 051176 004737 030264'  60$:  JSR      PC,CLRDN1   ; WRITE ONE TO CLEAR DNI
7607                ; ERROR ?
7608 051202 103010                BCC    70$              ; NO
7609 051204                FTL
      051204 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
7610 051210                ERRHRD  207.,ERR006,MSG003 ; YES, REPORT ERROR
      051210 104456                TRAP   C$ERRHRD
      051212 000317                .WORD  207
      051214 023030'                .WORD  ERR006
      051216 021736'                .WORD  MSG003
7611 051220                ESCAPE  TST              ; AND ABORT TEST
      051220 104410                TRAP   C$ESCAPE
      051222 001076                .WORD  L10047-.
7612                ;
7613                ;WRITE RING FORMAT
7614                ;
7615 051224 012705 012540'  70$:  MOV     #WTRNGS,R5      ; DEFAULT WRITE RING FORMAT FUNCTION
7616 051230 004737 032002'      JSR     PC,LDPCBB        ; LOAD FUNCTION -> PCBB
7617 051234 012705 012704'      MOV     #RFRMT,R5      ; DEFAULT RING FORMAT
7618 051240 012700 000006'      MOV     #6,R0          ; FORMAT = SIX WORDS
7619 051244 004737 032260'      JSR     PC,LDUDBB       ; LOAD RING FORMAT -> UDBB
7620 051250 012777 004100 126746 MOV     #DNI!INTE,@PCSR0
7621 051256 112777 000102 126740 MOVB   #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
7622 051264 004737 026550'      JSR     PC,CHKDNI      ; DNI ?
7623 051270 103010                BCC    80$              ; YES
7624 051272                FTL
      051272 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
7625 051276                ERRHRD  210.,ERR010,MSG003 ; NO, REPORT ERROR
      051276 104456                TRAP   C$ERRHRD
      051300 000322                .WORD  210
      051302 023331'                .WORD  ERR010
      051304 021736'                .WORD  MSG003
7626 051306                ESCAPE  TST              ; AND ABORT TEST
      051306 104410                TRAP   C$ESCAPE
      051310 001010                .WORD  L10047-.
7627                ;
7628 051312 004737 030264'  80$:  JSR      PC,CLRDN1   ; WRITE ONE TO CLEAR DNI
7629                ; ERROR ?
7630 051316 103010                BCC    90$              ; NO
7631 051320                FTL
  
```

```

051320 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
7632 051324      ERRHRD  211.,ERR006,MSG003 ; YES, REPORT ERROR
      051324      104456      TRAP      C#ERHRD
      051326      000323      .WORD    211
      051330      023030      .WORD    ERR006
      051332      021736'      .WORD    MSG003
7633 051334      ESCAPE  TST      ; AND ABORT TEST
      051334      104410      TRAP      C#ESCAPE
      051336      000762      .WORD    L10047-.
7634      ;
7635      ;WRITE PHYSICAL ADDRESS
7636
7637 051340      90$:
7638 051340      012705 000272'      MOV      #DEFAULT,R5      ; GET DEFAULT PHYSICAL ADDRESS
7639 051344      004737 032050'      JSR      PC,LDPHYA      ; SAVE IT IN DEFAULT TABLE
7640 051350      012705 012500'      MOV      #WTPHYA,R5      ; DEFAULT WRITE PHYSICAL ADDR FUNC
7641 051354      004737 032002'      JSR      PC,LDPCBB      ; LOAD FUNCTION -> PCBB
7642 051360      012777 004100 126636      MOV      #DNI!INTE,#PCSR0
7643 051366      112777 000102 126630      MOVB    #INTE!GETCMD,#PCSR0 ; ISSUE GET_CMD PORT COMMAND
7644 051374      004737 026550'      JSR      PC,CHKDNI      ; DNI ?
7645 051400      103010      BCC     100$      ; YES
7646 051402      FTL
051402 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
7647 051406      ERRHRD  212.,ERR010,MSG003 ; NO, REPORT ERROR
      051406      104456      TRAP      C#ERHRD
      051410      000324      .WORD    212
      051412      023331'      .WORD    ERR010
      051414      021736'      .WORD    MSG003
7648 051416      ESCAPE  TST      ; AND ABORT TEST
      051416      104410      TRAP      C#ESCAPE
      051420      000700      .WORD    L10047-.
7649      ;
7650 051422      004737 030264'      100$: JSR      PC,CLRDN1      ; WRITE ONE TO CLEAR DNI
7651      7651:
7652 051426      103010      BCC     110$      ; ERROR ?
7653 051430      FTL      ; NO
051430 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
7654 051434      ERRHRD  213.,ERR006,MSG003 ; YES, REPORT ERROR
      051434      104456      TRAP      C#ERHRD
      051436      000325      .WORD    213
      051440      023050'      .WORD    ERR006
      051442      021736'      .WORD    MSG003
7655 051444      ESCAPE  TST      ; AND ABORT TEST
      051444      104410      TRAP      C#ESCAPE
      051446      000652      .WORD    L10047-.
7656      ;
7657      ;SET UP RINGS FOR LOOPBACK
7658
7659 051450      110$: MOV      #TDRB1A,R5      ; DEFAULT ONE BUFFER TRANSMIT RING
7660 051454      004737 032164'      JSR      PC,LDTDRB      ; LOAD TDRB
7661 051460      012705 013050'      MOV      #RDRB2A,R5      ; DEFAULT CHAINED RECEIVE RING
7662 051464      004737 032070'      JSR      PC,LDRDRB      ; LOAD RDRB
    
```

```

7663                                     ;
7664                                     ;SET UP BUFFERS AND START
7665
7666 051470 005037 016562'             CLR      DOCRC                ; NO CRC
7667 051474 012737 000006 016560'     MOV      #6,BYTCNT          ; BYTES/PACKET
7668 051502 004737 033006'             JSR      PC,SETBUF          ; SET UP BUFFERS
7669 051506 012777 004100 126510     MOV      @DNI!INTE,@PCSR0
7670 051514 112777 000104 126502     MOVB    @INTE!START,@PCSR0 ; ISSUE START PORT COMMAND
7671 051522 004737 026550'             JSR      PC,CHKDNI          ; DNI?
7672 051526 103010                     BCC     120$                ; YES
7673 051530
                                     FTL
                                     ;
                                     ;
051530 004737 026652'             JSR      PC,CHKFTL          ; 'FATL' BIT SET?
7674 051534
051534 104456
051536 000326
051540 023447'
051542 021736'
ERRHRD 214.,ERR012,MSG003          ; NO, REPORT ERROR
                                     TRAP    C$ERHRD
                                     .WORD  214
                                     .WORD  ERR012
                                     .WORD  MSG003
7675 051544
051544 104410
051546 000552
ESCAPE TST                          ; AND ABORT TEST
                                     TRAP    C$ESCAPE
                                     .WORD  L10047-.
7676
7677 051550 004737 030264'           i 120$: JSR      PC,CLRDN1          ; WRITE ONE TO CLEAR DNI
7678
7679 051554 103010                     BCC     130$                ; ERROR ?
7680 051556
                                     FTL
                                     ; NO
                                     ;
051556 004737 026652'             JSR      PC,CHKFTL          ; 'FATL' BIT SET?
7681 051562
051562 104456
051564 000327
051566 023030'
051570 021736'
ERRHRD 215.,ERR006,MSG003          ; YES, REPORT ERROR
                                     TRAP    C$ERHRD
                                     .WORD  215
                                     .WORD  ERR006
                                     .WORD  MSG003
7682 051572
051572 104410
051574 000524
ESCAPE TST                          ; AND ABORT TEST
                                     TRAP    C$ESCAPE
                                     .WORD  L10047 .
7683
7684 051576 004737 027566'           i 130$: JSR      PC,CHKTXI          ; TXI ?
7685 051602 103010                     BCC     140$                ; YES
7686 051604
                                     FTL
                                     ;
051604 004737 026652'             JSR      PC,CHKFTL          ; 'FATL' BIT SET?
7687 051610
051610 104456
051612 000330
051614 023530'
051616 021736'
ERRHRD 216.,ERR013,MSG003          ; NO, REPORT ERROR
                                     TRAP    C$ERHRD
                                     .WORD  216
                                     .WORD  ERR013
                                     .WORD  MSG003
7688 051620
051620 104410
051622 000476
ESCAPE TST                          ; AND ABORT TEST
                                     TRAP    C$ESCAPE
                                     .WORD  L10047-.
7689
7690 051624 004737 030500'           i 140$: JSR      PC,CLR1TXI          ; WRITE ONE TO CLEAR TXI
7691
7692 051630 103010                     BCC     150$                ; ERROR ?
                                     ; NO

```

7693	051632			FTL				
	051632	004737	026652	JSR	PC,CHKFTL	:	'FATL' BIT SET?	
7694	051636			ERRHRD	217.,ERR014,MSG003	:	YES, REPORT ERROR	
	051636	104456				TRAP	C\$ERHRD	
	051640	000331				.WORD	217	
	051642	023561				.WORD	FRR014	
	051644	021736				.WORD	MSG003	
7695	051646			ESCAPE	TST	:	AND ABORT TEST	
	051646	104410				TRAP	C\$ESCAPE	
	051650	000450				.WORD	L10047-	
7696								
7697	051652	012705	000620	150\$:	MOV	#TDRB,R5	:	CHECK TDRB OWNERSHIP
7698	051656	004737	027024		JSR	PC,CHKOWN	:	OWN = PORT DRIVER ?
7699	051662	103010			BCC	160\$	:	YES
7700	051664				FTL			
	051664	004737	026652		JSR	PC,CHKFTL	:	'FATL' BIT SET?
7701	051670			ERRHRD	220.,ERR018	:	NO, REPORT ERROR	
	051670	104456				TRAP	C\$ERHRD	
	051672	000334				.WORD	220	
	051674	024026				.WORD	ERR018	
	051676	000000				.WORD	0	
7702	051700			ESCAPE	TST	:	AND ABORT TEST	
	051700	104410				TRAP	C\$ESCAPE	
	051702	000416				.WORD	L10047 .	
7703								
7704	051704	012705	016264	160\$:	MOV	#TDR14A,R5	:	POINT TO EXPECTED TDRB
7705	051710	004737	032370		JSR	PC,LDXTDR	:	LOAD INTO XTDRB0 TABLE
7706	051714	012705	000620		MOV	#TDRB,R5	:	CHECK TDRB
7707	051720	004737	027500		JSR	PC,CHKTDR	:	ERRORS ?
7708	051724	103010			BCC	170\$	:	NO
7709	051726				FTL			
	051726	004737	026652		JSR	PC,CHKFTL	:	'FATL' BIT SET?
7710	051732			ERRHRD	221.,ERR020,MSG005	:	YES, REPORT ERROR	
	051732	104456				TRAP	C\$ERHRD	
	051734	000335				.WORD	221	
	051736	024206				.WORD	ERR020	
	051740	022042				.WORD	MSG005	
7711	051742			ESCAPE	TST	:	AND ABORT TEST	
	051742	104410				TRAP	C\$ESCAPE	
	051744	000354				.WORD	L10047-.	
7712								
7713	051746	004737	027316	170\$:	JSR	PC,CHKRXI	:	RXI ?
7714	051752	103010			BCC	180\$	:	YES
7715	051754				FTL			
	051754	004737	026652		JSR	PC,CHKFTL	:	'FATL' BIT SET?
7716	051760			ERRHRD	222.,ERR015,MSG003	:	NO, REPORT ERROR	
	051760	104456				TRAP	C\$ERHRD	
	051762	000336				.WORD	222	
	051764	023627				.WORD	ERR015	

```

051766 021736'
7717 051770          ESCAPE TST          ; AND ABORT TEST          .WORD  MSG003
    051770 104410
    051772 000326          TRAP      C$ESCAPE
7718          ;
7719 051774 004737 030432' 180$: JSR    PC,CLRRXI          ; WRITE ONE TO CLEAR RXI
7720          ; ERROR ?
7721 052000 103010          BCC    190$          ; NO
7722 052002          FTL
    052002 004737 026652'          JSR    PC,CHKFTL          ; 'FATL' BIT SET?
7723 052006          ERRHRD 223.,ERR016,MSG003 ; YES, REPORT ERROR
    052006 104456          TRAP      C$ERHRD
    052010 000337          .WORD    223
    052012 023660'          .WORD    ERR016
    052014 021736'          .WORD    MSG003
7724 052016          ESCAPE TST          ; AND ABORT TEST          TRAP      C$ESCAPE
    052016 104410          .WORD    L10047-.
    052020 000300
7725          ;CHECK FIRST RING ENTRY
7726
7727
7728 052022 012705 000660' 190$: MOV    #RDRB,R5          ; CHECK RDRB OWNERSHIP
7729 052026 004737 027024' JSR    PC,CHKOWN          ; OWN = PORT DRIVER ?
7730 052032 103010          BCC    200$          ; YES
7731 052034          FTL
    052034 004737 026652'          JSR    PC,CHKFTL          ; 'FATL' BIT SET?
7732 052040          ERRHRD 224.,ERR030          ; NO, REPORT ERROR
    052040 104456          TRAP      C$ERHRD
    052042 000340          .WORD    224
    052044 024753'          .WORD    ERR030
    052046 000000          .WORD    0
7733 052050          ESCAPE TST          ; AND ABORT TEST          TRAP      C$ESCAPE
    052050 104410          .WORD    L10047-.
    052052 000246
7734          ;
7735 052054 012705 016414' 200$: MOV    #RDR17A,R5          ; POINT TO EXPECTED RDRB
7736 052060 004737 032340' JSR    PC,LDXRDR          ; LOAD INTO XRDRBO TABLE
7737 052064 012705 000660' MOV    #RDRB,R5          ; CHECK RDRB
7738 052070 004737 027206' JSR    PC,CHKRDR          ; ERRORS ?
7739 052074 103010          BCC    210$          ; NO
7740 052076          FTL
    052076 004737 026652'          JSR    PC,CHKFTL          ; 'FATL' BIT SET?
7741 052102          ERRHRD 225.,ERR036,MSG006 ; YES, REPORT ERROR
    052102 104456          TRAP      C$ERHRD
    052104 000341          .WORD    225
    052106 025345'          .WORD    ERR036
    052110 022204'          .WORD    MSG006
7742 052112          ESCAPE TST          ; AND ABORT TEST          TRAP      C$ESCAPE
    052112 104410          .WORD    L10047 .
    052114 000204
7743
  
```

```

7744          ;CHECK SECOND RING ENTRY
7745
7746 052116 012705 016424      210$:  MOV    #RDR17B,R5      ; POINT TO EXPECTED RDRB
7747 052122 004737 032340'    JSR    PC,LDXRDR      ; LOAD INTO XRDRBO TABLE
7748 052126 012705 000670'    MOV    #RDRB+8.,R5   ; CHECK RDRB
7749 052132 004737 027206'    JSR    PC,CHKRDR     ; ERRORS ?
7750 052136 103010             BCC    230$          ; NO
7751 052140             FTL

          052140 004737 026652'    JSR    PC,CHKFTL     ; 'FATL' BIT SET?

7752 052144             ERRHRD 226.,ERR037,MSG006      ; YES, REPORT ERROR
          052144 104456             TRAP    C$ERHRD
          052146 000342             .WORD  226
          052150 025433'           .WORD  ERR037
          052152 022204'           .WORD  MSG006

7753 052154             ESCAPE  TST                    ; AND ABORT TEST
          052154 104410             TRAP    C$ESCAPE
          052156 000142             .WORD  L10047-.

7754
7755 052160 012777 004100 126036 ; 230$:  MOV    #DNI!INTE,@PCSRO
7756 052166 112777 000117 126030  MOVB   #INTE!STOP,@PCSRO ; ISSUE STOP PORT COMMAND
7757 052174 004737 026550'    JSR    PC,CHKDNI     ; DNI ?
7758 052200 103010             BCC    240$          ; YES
7759 052202             FTL

          052202 004737 026652'    JSR    PC,CHKFTL     ; 'FATL' BIT SET?

7760 052206             ERRHRD 227.,ERR019,MSG003      ; NO, REPORT ERROR
          052206 104456             TRAP    C$ERHRD
          052210 000343             .WORD  227
          052212 024126'           .WORD  ERR019
          052214 021736'           .WORD  MSG003

7761 052216             ESCAPE  TST                    ; AND ABORT TEST
          052216 104410             TRAP    C$ESCAPE
          052220 000100             .WORD  L10047-.

7762
7763 052222 004737 030264'    ; 240$:  JSR    PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
7764             BCC    250$          ; ERROR ?
7765 052226 103010             ; NO
7766 052230             FTL

          052230 004737 026652'    JSR    PC,CHKFTL     ; 'FATL' BIT SET?

7767 052234             ERRHRD 230.,ERR006,MSG003      ; YES, REPORT ERROR
          052234 104456             TRAP    C$ERHRD
          052236 000346             .WORD  230
          052240 023030'           .WORD  ERR006
          052242 021736'           .WORD  MSG003

7768 052244             ESCAPE  TST                    ; AND ABORT TEST
          052244 104410             TRAP    C$ESCAPE
          052246 000052             .WORD  L10047-.

7769
7770 052250 004737 030346'    ; 250$:  JSR    PC,CLINTR      ; INSURE DELUA INTR BITS CLEAR
7771             EXIT    TST
7772 052254 104432             TRAP    C$EXIT
  
```

.WORD L10047-.

052256 000042  
7773  
7774 ;LOCAL TEST MESSAGE  
7775  
7776 052260 104 105 114 T17ID: .ASCIZ 'DELUA DISABLE RECEIVE CHAINING '  
052263 125 101 040  
052266 104 111 123  
052271 101 102 114  
052274 105 040 122  
052277 105 103 105  
052302 111 126 105  
052305 040 103 110  
052310 101 111 116  
052313 111 116 107  
052316 040 000  
7777 .EVEN  
7778  
7779 052320 ENDTST  
052320  
052320 104401

L10047: TRAP C\$ETST



7781  
 7782  
 7783  
 7784  
 7785  
 7786  
 7787  
 7788  
 7789  
 7790  
 7791  
 7792  
 7793  
 7794  
 7795  
 7796  
 7797  
 7798  
 7799  
 7800  
 7801

.SBTTL TEST 18: TRANSMIT CHAINING ERROR TEST

```

*****
:
: THIS TEST VERIFIES THAT A TRANSMIT BUFL ERROR CAN BE GENERATED.
: AN INTERNAL LOOPBACK IS ATTEMPTED WITH TRANSMIT BUFFERS CHAINED
: AND SUCCESSIVE OWNED RINGS HAVING STP SET.
: A BUFL ERROR IS EXPECTED IN THE TRANSMIT DESCRIPTOR RING.
:
: TEST SEQUENCE:
: 1. WRITE MODE REGISTER = INTERNAL LOOPBACK AND PROM MODE
: 2. WRITE RING FORMAT
: 3. WRITE PHYSICAL ADDRESS
: 4. SET UP PINGS AND BUFFERS
:    TRANSMIT RING = CHAINED WITH SUCCESSIVE STPs
: 5. ISSUE START
: 6. CHECK FOR BUFL ERROR IN TORB+6
: 7. ISSUE STOP
:
*****
  
```

7802 052322  
 052322  
 7803  
 7804 052322

BGNTST

T18::

PNTMAC T18ID

052322 012704 053552'  
 052326 004737 032734'

```

MOV    #T18ID,R4      ;GET POINTER TO TEST NAME MESSAGE
JSR    PC,PNTID       ;PRINT TEST NUMBER AND NAME
  
```

: END OF MACRO EXPANSION OF 'PNTMAC'

7805 052332 004737 033434'  
 7806 052336 103034  
 7807 052340 012777 004100 125656  
 7808 052346 112777 000140 125650  
 7809 052354 004737 027736'  
 7810 052360 103010  
 7811 052362

```

JSR    PC,TINIT      ; IS A DEVICE RESET NEEDED?
BCC    30$           ; NO
MOV    #DNI!INTE,@PCSR0 ; PRECONDITION INTR EN.
MOVB   #INTE!RSET,@PCSR0 ; YES, RESET DELUA
JSR    PC,CKDNI      ; DNI ?
BCC    20$           ; YES
FTL
  
```

052362 004737 026652'

```

JSR    PC,CHKFTL     ; 'FATL' BIT SET?
  
```

7812 052366  
 052366 104456  
 052370 000347  
 052372 023030'  
 052374 021736'

```

ERRHRD 231.,ERR006,MSG003 ; NO, REPORT ERROR
  
```

```

TRAP   C$ERRHRD
.WORD  231
.WORD  ERR006
.WORD  MSG003
  
```

7813 052376  
 052376 104410  
 052400 001212

```

ESCAPE TST           ; AND ABORT TEST
  
```

```

TRAP   C$ESCAPE
.WORD  L10050-.
  
```

7814  
 7815 052402 004737 030264'  
 7816  
 7817 052406 103010  
 7818 052410

: 20\$:

```

JSR    PC,CLRDNIS   ; WRITE ONE TO CLEAR DNI
; ERROR ?
BCC    30$           ; NO
FTL
  
```

052410 004737 026652'

```

JSR    PC,CHKFTL     ; 'FATL' BIT SET?
  
```

```

7819 052414          ERKHRD 232.,ERR006,MSG003      ; YES, REPORT ERROR
      052414 104456
      052416 000350
      052420 023030'
      052422 021736'
7820 052424          ESCAPE TST                    ; AND ABORT TEST
      052424 104410
      052426 001164
7821
7822 052430          ;
7823 052430 004737 030212'          ; 30$: JSR PC,CLRBUF          ; CLEAR XMIT,RCV BUFFERS
7824 052434 004737 031732'          ; JSR PC,LDDFLT          ; LOAD DEFAULT PHY.ADDRESS TABLES
7825 052440 004737 032032'          ; JSR PC,LDPCSR          ; ADDRESS OF PCBB -> PCSR2!3
7826 052444 012777 004100 125552    ; MOV #DNI!INTE,@PCSR0   ; ENABLE INTERRUPTS
7827 052452 112777 000101 125544    ; MOVB #INTE!GETPCB,@PCSR0 ; ISSUE GET_PCBB PORT COMMAND
7828 052460 004737 026550'          ; JSR PC,CHKDNI          ; DNI?
7829 052464 103010
7830 052466          BCC 40$                    ; YES
      FTL
      052466 004737 026652'          JSR PC,CHKFTL          ; 'FATL' BIT SET?
7831 052472          ERRHRD 233.,ERR009,MSG003      ; NO, REPORT ERROR
      052472 104456
      052474 000351
      052476 023245'
      052500 021736'
7832 052502          ESCAPE TST                    ; AND ABORT TEST
      052502 104410
      052504 001106
7833
7834 052506 004737 030264'          ; 40$: JSR PC,CLRDN1          ; WRITE ONE TO CLEAR DNI
7835
7836 052512 103010
7837 052514          BCC 50$                    ; NO
      FTL
      052514 004737 026652'          JSR PC,CHKFTL          ; 'FATL' BIT SET?
7838 052520          ERRHRD 234.,ERR006,MSG003      ; YES, REPORT ERROR
      052520 104456
      052522 000352
      052524 023030'
      052526 021736'
7839 052530          ESCAPE TST                    ; AND ABORT TEST
      052530 104410
      052532 001060
7840
7841          ;WRITE MODE REGISTER = INTERNAL LOOPBACK AND PROM MODE
7842
7843 052534 012705 012600'          ; 50$: MOV #WMODE,R5          ; DEFAULT WRITE MODE FUNCTION
7844 052540 004737 032002'          ; JSR PC,LDPCCBB          ; LOAD FUNCTION -> PCBB
7845 052544 012777 004100 125452    ; MOV #DNI!INTE,@PCSR0   ; ENABLE INTERRUPTS
7846 052552 112777 000102 125444    ; MOVB #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
7847 052560 004737 026550'          ; JSR PC,CHKDNI          ; DNI ?
7848 052564 103010
7849 052566          BCC 60$                    ; YES
      FTL
      052566 004737 026652'          JSR PC,CHKFTL          ; 'FATL' BIT SET?
  
```

```

7850 052572          ERRHRD 235.,ERR010,MSG003      ; NO, REPORT ERROR
      052572 104456
      052574 000353
      052576 023331'
      052600 021736'
7851 052602          ESCAPE TST                    ; AND ABORT TEST
      052602 104410
      052604 001006
7852
7853 052606 004737 030264'      ; 60$: JSR PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
7854
7855 052612 103010
7856 052614          BCC 70$                       ; ERROR ?
      FTL
      JSR PC,CHKFTL      ; 'FATL' BIT SET?
7857 052620          ERRHRD 236.,ERR006,MSG003      ; YES, REPORT ERROR
      052620 104456
      052622 000354
      052624 023030'
      052626 021736'
7858 052630          ESCAPE TST                    ; AND ABORT TEST
      052630 104410
      052632 000760
7859
7860          ;WRITE RING FORMAT
7861 052634 012705 012540'      ; 70$: MOV #WTRNGS,R5      ; DEFAULT WRITE RING FORMAT FUNCTION
7862 052640 004737 032002'      ; JSR PC,LDPCCB          ; LOAD FUNCTION -> PCBB
7863 052644 012705 012704'      ; MOV #RFRMT,R5         ; DEFAULT RING FORMAT
7864 052650 012700 000006      ; MOV #6,R0             ; FORMAT = SIX WORDS
7865 052654 004737 032260'      ; JSR PC,LDUDBB         ; LOAD RING FORMAT -> UDBB
7866 052660 012777 004100 125336 ; MOV #DNI!INTE,@PCSR0  ; ENABLE INTERRUPTS
7867 052666 112777 000102 125330 ; MOVB #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
7868 052674 004737 026550'      ; JSR PC,CHKDNI        ; DNI ?
7869 052700 103010
7870 052702          BCC 80$                       ; YES
      FTL
      JSR PC,CHKFTL      ; 'FATL' BIT SET?
7871 052706          ERRHRD 237.,ERR010,MSG003      ; NO, REPORT ERROR
      052706 104456
      052710 000355
      052712 023331'
      052714 021736'
7872 052716          ESCAPE TST                    ; AND ABORT TEST
      052716 104410
      052720 000672
7873
7874 052722 004737 030264'      ; 80$: JSR PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
7875
7876 052726 103010
7877 052730          BCC 90$                       ; ERROR ?
      FTL
      JSR PC,CHKFTL      ; 'FATL' BIT SET?
7878 052734          ERRHRD 240.,ERR006,MSG003      ; YES, REPORT ERROR
    
```

```

052734 104456
052736 000360
052740 023030'
052742 021736'
7879 052744          ESCAPE TST          ; AND ABORT TEST
052744 104410
052746 000644
7880
7881 ;WRITE PHYSICAL ADDRESS
7882
7883 052750          90$:
7884 052750 012705 000272'          MOV    #DEFAULT,R5          ; GET DEFAULT PHYSICAL ADDRESS
7885 052754 004737 032050'          JSR    PC,LDPHYA          ; SAVE IT IN DEFAULT TABLE
7886 052760 012705 012500'          MOV    #WTPHYA,R5        ; DEFAULT WRITE PHYSICAL ADDR FUNC
7887 052764 004737 032002'          JSR    PC,LDPCBB         ; LOAD FUNCTION -> PCBB
7888 052770 012777 004100 125226     MOV    #DNI!INTE,@PCSR0  ; ENABLE INTERRUPTS
7889 052776 112777 000102 125220     MOVB  #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
7890 053004 004737 026550'          JSR    PC,CHKDNI        ; DNI ?
7891 053010 103010
7892 053012          BCC    100$             ; YES
7892 053012 004737 026652'          JSR    PC,CHKFTL        ; 'FATL' BIT SET?
7893 053016          ERRHRD 241.,ERR010,MSG003 ; NO, REPORT ERROR
053016 104456
053020 000361
053022 023331'
053024 021736'
7894 053026          ESCAPE TST          ; AND ABORT TEST
053026 104410
053030 000562
7895
7896 053032 004737 030264'          ;100$: JSR    PC,CLRDN1    ; WRITE ONE TO CLEAR DNI
7897
7898 053036 103010          BCC    110$             ; ERROR ?
7899 053040          FTL
7899 053040          JSR    PC,CHKFTL        ; 'FATL' BIT SET?
7900 053044          ERRHRD 242.,ERR006,MSG003 ; YES, REPORT ERROR
053044 104456
053046 000362
053050 023030'
053052 021736'
7901 053054          ESCAPE TST          ; AND ABORT TEST
053054 104410
053056 000534
7902
7903 ;SET UP RINGS
7904 053060 012705 014550'          ;110$: MOV    #TDRB1E,R5    ; DEFAULT ERROR TRANSMIT RING
7905 053064 004737 032164'          JSR    PC,LDTDRB        ; LOAD TDRB
7906 053070 012705 012750'          MOV    #RDRB1A,R5      ; DEFAULT ONE BUFFER RECEIVE RING
7907 053074 004737 032070'          JSR    PC,LDRDRB        ; LOAD RDRB
7908
7909 ;SET UP BUFFERS AND START
7910
7911 053100 005037 016562'          CLR    D0CRC           ; NO CRC
  
```

TRAP C\$ERHRD  
 .WORD 240  
 .WORD ERR006  
 .WORD MSG003

TRAP C\$ESCAPE  
 .WORD L10050-.

TRAP C\$ERHRD  
 .WORD 241  
 .WORD ERR010  
 .WORD MSG003

TRAP C\$ESCAPE  
 .WORD L10050-.

TRAP C\$ERHRD  
 .WORD 242  
 .WORD ERR006  
 .WORD MSG003

TRAP C\$ESCAPE  
 .WORD L10050-.

```

7912 053104 012737 000006 016560'   MOV    #6,BYTCNT      ; BYTES/PACKET
7913 053112 004737 033006'   JSR    PC,SETBUF     ; SET UP BUFFERS
7914 053116 012777 004100 125100   MOV    #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
7915 053124 112777 000104 125072   MOVB   #INTE!START,@PCSR0 ; ISSUE START PORT COMMAND
7916 053132 004737 026550'   JSR    PC,CHKDNI    ; DNI?
7917 053136 103010                BCC    120$         ; YES
7918 053140                FTL

      053140 004737 026652'   JSR    PC,CHKFTL    ; 'FATL' BIT SET?

7919 053144                ERRHRD 243.,ERR012,MSG003 ; NO, REPORT ERROR
      053144 104456                TRAP  C$ERHRD
      053146 000363                .WORD 243
      053150 023447'                .WORD ERR012
      053152 021736'                .WORD MSG003

7920 053154                ESCAPE TST          ; AND ABORT TEST
      053154 104410                TRAP  C$ESCAPE
      053156 000434                .WORD L10050-.

7921                ;
7922 053160 004737 030264'   JSR    PC,CLRDN1    ; WRITE ONE TO CLEAR DNI
7923                ; ERROR ?
7924 053164 103010                BCC    130$         ; NO
7925 053166                FTL

      053166 004737 026652'   JSR    PC,CHKFTL    ; 'FATL' BIT SET?

7926 053172                ERRHRD 245.,ERR006,MSG003 ; YES, REPORT ERROR
      053172 104456                TRAP  C$ERHRD
      053174 000365                .WORD 245
      053176 023030'                .WORD ERR006
      053200 021736'                .WORD MSG003

7927 053202                ESCAPE TST          ; AND ABORT TEST
      053202 104410                TRAP  C$ESCAPE
      053204 000406                .WORD L10050-.

7928                ;
7929 053206 004737 027566'   JSR    PC,CHKTXI    ; TXI ?
7930 053212 103010                BCC    140$         ; YES
7931 053214                FTL

      053214 004737 026652'   JSR    PC,CHKFTL    ; 'FATL' BIT SET?

7932 053220                ERRHRD 246.,ERR013,MSG003 ; NO, REPORT ERROR
      053220 104456                TRAP  C$ERHRD
      053222 000366                .WORD 246
      053224 023530'                .WORD ERR013
      053226 021736'                .WORD MSG003

7933 053230                ESCAPE TST          ; AND ABORT TEST
      053230 104410                TRAP  C$ESCAPE
      053232 000360                .WORD L10050-.

7934                ;
7935 053234 004737 030500'   JSR    PC,CLRTXI    ; WRITE ONE TO CLEAR TXI
7936                ; ERROR ?
7937 053240 103010                BCC    150$         ; NO
7938 053242                FTL

      053242 004737 026652'   JSR    PC,CHKFTL    ; 'FATL' BIT SET?
  
```

```

7939 053246          ERRHRD  247.,ERR014,MSG003      ; YES, REPORT ERROR
      053246 104456
      053250 000367
      053252 023561'
      053254 021736'
7940 053256          ESCAPE  TST                    ; AND ABORT TEST
      053256 104410
      053260 000332
7941
7942          ;CHECK FIRST RING ENTRY
7943
7944 053262 012705 000620' 150$: MOV    #TDRB,R5          ; CHECK TDRB OWNERSHIP
7945 053266 004737 027024'   JSR    PC,CHKOWN      ; OWN = PORT DRIVER ?
7946 053272 103010          BCC    160$           ; YES
7947 053274          FTL
      053274 004737 026652'   JSR    PC,CHKFTL      ; 'FATL' BIT SET?
7948 053300          ERRHRD  250.,ERR027            ; NO, REPORT ERROR
      053300 104456
      053302 000372
      053304 024540'
      053306 000000
7949 053310          ESCAPE  TST                    ; AND ABORT TEST
      053310 104410
      053312 000300
7950
7951 053314 012705 016304' 160$: MOV    #TDR18A,R5      ; POINT TO EXPECTED TDRB
7952 053320 004737 032370'   JSR    PC,LDXTDR     ; LOAD INTO XTDRBO TABLE
7953 053324 012705 000620'   MOV    #TDRB,R5      ; CHECK TDRB
7954 053330 004737 027500'   JSR    PC,CHKTDR     ; ERRORS ?
7955 053334 103010          BCC    162$           ; NO
7956 053336          FTL
      053336 004737 026652'   JSR    PC,CHKFTL      ; 'FATL' BIT SET?
7957 053342          ERRHRD  251.,ERR033,MSG005      ; YES, REPORT ERROR
      053342 104456
      053344 000373
      053346 025166'
      053350 022042'
7958 053352          ESCAPE  TST                    ; AND ABORT TEST
      053352 104410
      053354 000236
7959
7960          ;CHECK SECOND RING ENTRY
7961
7962 053356 012705 000634' 162$: MOV    #TDRB+12.,R5    ; CHECK TDRB OWNERSHIP
7963 053362 004737 027024'   JSR    PC,CHKOWN      ; OWN = PORT DRIVER ?
7964 053366 103010          BCC    164$           ; YES
7965 053370          FTL
      053370 004737 026652'   JSR    PC,CHKFTL      ; 'FATL' BIT SET?
7966 053374          ERRHRD  252.,ERR028            ; NO, REPORT ERROR
      053374 104456
      053376 000374
  
```

TRAP C#ERHRD  
 .WORD 247  
 .WORD ERR014  
 .WORD MSG003

TRAP C#ESCAPE  
 .WORD L10050-

TRAP C#ERHRD  
 .WORD 250  
 .WORD ERR027  
 .WORD 0

TRAP C#ESCAPE  
 .WORD L10050-

TRAP C#ERHRD  
 .WORD 251  
 .WORD ERR033  
 .WORD MSG005

TRAP C#ESCAPE  
 .WORD L10050-

TRAP C#ERHRD  
 .WORD 252

```

053400 024645' .WORD ERR028
053402 000000 .WORD 0
7967 053404 ESCAPE TST ; AND ABORT TEST TRAP C#ESCAPE
053404 104410 .WORD L10050-.
053406 000204
7968
7969 053410 012705 016314' ; 164: MOV #TDR18B,R5 ; POINT TO EXPECTED TDRB
7970 053414 004737 032370' JSR PC,LDXTDR ; LOAD INTO XTDRB0 TABLE
7971 053420 012705 000630' MOV #TDRB+8.,R5 ; CHECK TDRB
7972 053424 004737 027500' JSR PC,CHKTDR ; ERRORS ?
7973 053430 103010 BCC 230# ; NO
7974 053432 FTL
053432 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?
7975 053436 ERRHRD 253.,ERR034,MSG005 ; YES, REPORT ERROR TRAP C#ERHRD
053436 104456 .WORD 253
053440 000375 .WORD ERR034
053442 025255' .WORD MSG005
053444 022042'
7976 053446 ESCAPE TST ; AND ABORT TEST TRAP C#ESCAPE
053446 104410 .WORD L10050-.
053450 000142
7977
7978 053452 012777 004100 124544 ; 230: MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
7979 053460 112777 000117 124536 MOVB #INTE!STOP,@PCSR0 ; ISSUE STOP PORT COMMAND
7980 053466 004737 026550' JSR PC,CHKDNI ; DNI ?
7981 053472 103010 BCC 240# ; YES
7982 053474 FTL
053474 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?
7983 053500 ERRHRD 254.,ERR019,MSG003 ; NO, REPORT ERROR TRAP C#ERHRD
053500 104456 .WORD 254
053502 000376 .WORD ERR019
053504 024126' .WORD MSG003
053506 021736'
7984 053510 ESCAPE TST ; AND ABORT TEST TRAP C#ESCAPE
053510 104410 .WORD L10050-.
053512 000100
7985
7986 053514 004737 030264' ; 240.: JSR PC,CLRDN1 ; WRITE ONE TO CLEAR DNI
7987 BCC 250# ; ERROR ?
7988 053520 103010 FTL ; NO
7989 053522
053522 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?
7990 053526 ERRHRD 255.,ERR006,MSG003 ; YES, REPORT ERROR TRAP C#ERHRD
053526 104456 .WORD 255
053530 000377 .WORD ERR006
053532 023030' .WORD MSG003
053534 021736'
7991 053536 ESCAPE TST ; AND ABORT TEST TRAP C#ESCAPE
053536 104410 .WORD L10050-.
053540 000052
7992 053542 ; 250:
    
```

```
7993 053542 004737 030346'      JSR    PC,CLINTR      ; INSURE DELUA INTR BITS CLEAR
7994
7995 053546      EXIT    TST
      053546 104432      TRAP   C#EXIT
      053550 000042      .WORD  L10050-.
7996
7997      ;LOCAL TEST MESSAGE
7998
7999 053552      104      105      114  T18ID: .ASCIZ 'DELUA TRANSMIT CHAINING ERROR '
      053555      125      101      040
      053560      124      122      101
      053563      116      123      115
      053566      111      124      040
      053571      103      110      101
      053574      111      116      111
      053577      116      107      040
      053602      105      122      122
      053605      117      122      040
      053610      000
8000      .EVEN
8001
8002 053612      ENDTST
      053612      L10050:
      053612 104401      TRAP   C#ETST
```





```

8042 053706          ERRHRD 257.,ERR006,MSG003      ; YES, REPORT ERROR
      053706 104456
      053710 000401
      053712 023030'
      053714 021736'
      8043 053716          ESCAPE TST              ; AND ABORT TEST
      053716 104410
      53720 001502
      8044
      8045 053722          ; 30$:
      8046 053722 004737 030212'      JSR    PC,CLRBUF      ; CLEAR XMIT,RECV BUFFERS
      8047 053726 004737 031732'      JSR    PC,LDDFLT     ; LOAD DEFAULT PHY.ADDRESS TABLES
      8048 053732 004737 032032'      JSR    PC,LDPCSR     ; ADDRESS OF PCBB -> PCSR2!3
      8049 053736 012777 004100 124260  MOV    #DNI!INTF,@PCSR0 ; ENABLE INTERRUPTS
      8050 053744 112777 000101 124252  MOVB   #INTE!GETPCB,@PCSR0 ; ISSUE GET_PCBB PORT COMMAND
      8051 053752 004737 026550'      JSR    PC,CHKDNI     ; DNI?
      8052 053756 103010
      8053 053760          BCC    40$           ; YES
      FTL
      053760 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?
      8054 053764          ERRHRD 260.,ERR009,MSG003      ; NO, REPORT ERROR
      053764 104456
      053766 000404
      053770 023245'
      053772 021736'
      8055 053774          ESCAPE TST              ; AND ABORT TEST
      053774 104410
      053776 001424
      8056
      8057 054000 004737 030264'      ; 40$:
      8058
      8059 054004 103010
      8060 054006          BCC    50$           ; NO
      FTL
      054006 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?
      8061 054012          ERRHRD 261.,ERR006,MSG003      ; YES, REPORT ERROR
      054012 104456
      054014 000405
      054016 023030'
      054020 021736'
      8062 054022          ESCAPE TST              ; AND ABORT TEST
      054022 104410
      054024 001376
      8063
      8064 054026          ; 50$:
      8065
      8066 054026 012705 012600'      ;WRITE MODE REGISTER = INTERNAL LOOPBACK, CRC, AND PROM MODE
      8067 054032 004737 032002'      MOV    #WTMODE,R5    ; DEFAULT WRITE MODE FUNCTION
      8068 054036 012777 004100 124160  JSR    PC,LDPCCBB    ; LOAD FUNCTION -> PCBB
      8069 054044 112777 000102 124152  MOV    #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
      8070 054052 004737 026550'      MOVB   #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
      8071 054056 103010
      8072 054060          JSR    PC,CHKDNI     ; DNI ?
      BCC    60$           ; YES
      FTL
      054060 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?
  
```

```

8073 054064          ERRHRD 262.,ERR010,MSG003      ; NO, REPORT ERROR
      054064 104456
      054066 000406
      054070 023331'
      054072 021736'
      8074 054074          ESCAPE TST                ; AND ABORT TEST
      054074 104410
      054076 001324
      8075
      8076 054100 004737 030264'      ; 60$: JSR PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
      8077
      8078 054104 103010
      8079 054106
      054106 004737 026652'          ; BCC 70$                ; ERROR ?
      054106 004737 026652'          ; FTL                    ; NO
      054106 004737 026652'          ; JSR PC,CHKFTL         ; 'FATL' BIT SET?
      8080 054112          ERRHRD 263.,ERR006,MSG003      ; YES, REPORT ERROR
      054112 104456
      054114 000407
      054116 023030'
      054120 021736'
      8081 054122          ESCAPE TST                ; AND ABORT TEST
      054122 104410
      054124 001276
      8082
      8083
      8084 054126 012705 012540'      ; WRITE RING FORMAT
      8085 054132 004737 032002'      ; 70$: MOV #WTRNGS,R5    ; DEFAULT WRITE RING FORMAT FUNCTION
      8086 054136 012705 012704'      ; JSR PC,LDPCBB         ; LOAD FUNCTION -> PCBB
      8087 054142 012700 000006      ; MOV #RFRMT,R5        ; DEFAULT RING FORMAT
      8088 054146 004737 032260'      ; MOV #6,R0            ; FORMAT = SIX WORDS
      8089 054152 012777 004100 124044 ; JSR PC,LDUDBB         ; LOAD RING FORMAT -> UDBB
      8090 054160 112777 000102 124036 ; MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
      8091 054166 004737 026550'      ; MOVB #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
      8092 054172 103010
      8093 054174
      054174 004737 026652'          ; JSR PC,CHKFTL         ; 'FATL' BIT SET?
      8094 054200          ERRHRD 264.,ERR010,MSG003      ; NO, REPORT ERROR
      054200 104456
      054202 000410
      054204 023331'
      054206 021736'
      8095 054210          ESCAPE TST                ; AND ABORT TEST
      054210 104410
      054212 001210
      8096
      8097 054214 004737 030264'      ; 80$: JSR PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
      8098
      8099 054220 103010
      8100 054222
      054222 004737 026652'          ; BCC 90$                ; ERROR ?
      054222 004737 026652'          ; FTL                    ; NO
      054222 004737 026652'          ; JSR PC,CHKFTL         ; 'FATL' BIT SET?
      8101 054226          ERRHRD 265.,ERR006,MSG003      ; YES, REPORT ERROR
  
```

```

054226 104456
054230 000411
054232 023030'
054234 021736'
8102 054236          ESCAPE TST          ; AND ABORT TEST
054236 104410
054240 001162
8103
8104          ;WRITE PHYSICAL ADDRESS
8105
8106 054242
8107 054242 012705 000272'
8108 054246 004737 032050'
8109 054252 012705 012500'
8110 054256 004737 032002'
8111 054262 012777 004100 123734
8112 054270 112777 000102 123726
8113 054276 004737 026550'
8114 054302 103010
8115 054304          FTL
          054304 004737 026652'
          JSR      PC,CHKFTL          ; 'FATL' BIT SET?
8116 054310          ERRHRD 266.,ERR010,MSG003 ; NO, REPORT ERROR
054310 104456
054312 000412
054314 023331'
054316 021736'
8117 054320          ESCAPE TST          ; AND ABORT TEST
054320 104410
054322 001100
8118
8119 054324 004737 030264'
8120          ;100$: JSR      PC,CLRDNI          ; WRITE ONE TO CLEAR DNI
8121 054330 103010          BCC      110$          ; ERROR ?
8122 054332          FTL          ; NO
          054332 004737 026652'
          JSR      PC,CHKFTL          ; 'FATL' BIT SET?
8123 054336          ERRHRD 267.,ERR006,MSG003 ; YES, REPORT ERROR
054336 104456
054340 000413
054342 023030'
054344 021736'
8124 054346          ESCAPE TST          ; AND ABORT TEST
054346 104410
054350 001052
8125
8126          ;SET UP RINGS FOR TWO BUFFERS CHAINED LOOPBACK
8127
8128 054352 012705 015570'
8129 054356 004737 032164'
8130 054362 012705 013540'
8131 054366 004737 032070'
8132
8133          ;SET UP BUFFERS AND START
8134
          TRAP      C$ERHRD
          .WORD     265
          .WORD     ERR006
          .WORD     MSG003
          TRAP      C$ESCAPE
          .WORD     L10051-.
          TRAP      C$ERHRD
          .WORD     266
          .WORD     ERR010
          .WORD     MSG003
          TRAP      C$ERHRD
          .WORD     267
          .WORD     ERR006
          .WORD     MSG003
          TRAP      C$ESCAPE
          .WORD     L10051-.
          ;110$: MOV      #TDRB4A,R5          ; DEFAULT TWO BUFFER TRANSMIT RING
          JSR      PC,LDTDRB          ; LOAD TDRB
          MOV      #RDRB4A,R5          ; DEFAULT TWO BUFFER RECEIVE RING
          JSR      PC,LDRDRB          ; LOAD RDRB
  
```

8135	054372	005037	016562'		CLR	DOCRC		; NO CRC		
8136	054376	012737	000000	016560'	MOV	#0,BYTCNT		; BYTES/PACKET		
8137	054404	004737	033006'		JSR	PC,SETBUF		; SET UP BUFFERS		
8138	054410	012777	004100	123606	MOV	#DNI!INTE,@PCSR0		; ENABLE INTERRUPTS		
8139	054416	112777	000104	123600	MOVB	#INTE!START,@PCSR0		; ISSUE START PORT COMMAND		
8140	054424	004737	026550'		JSR	PC,CHKDNI		; DNI?		
8141	054430	103010			BCC	120\$		; YES		
8142	054432				FTL					
	054432	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
8143	054436				ERRHRD	270.,ERR012,MSG003		; NO, REPORT ERROR		
	054436	104456							TRAP	C\$ERHRD
	054440	000416							.WORD	270
	054442	023447'							.WORD	ERR012
	054444	021736'							.WORD	MSG003
8144	054446				ESCAPE	TST		; AND ABORT TEST		
	054446	104410							TRAP	C\$ESCAPE
	054450	000752							.WORD	L10051-
8145										
8146	054452	004737	030264'		JSR	PC,CLRDN1		; WRITE ONE TO CLEAR DNI		
8147								; ERROR ?		
8148	054456	103010			BCC	130\$		; NO		
8149	054460				FTL					
	054460	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
8150	054464				ERRHRD	271.,ERR006,MSG003		; YES, REPORT ERROR		
	054464	104456							TRAP	C\$ERHRD
	054466	000417							.WORD	271
	054470	023030'							.WORD	ERR006
	054472	021736'							.WORD	MSG003
8151	054474				ESCAPE	TST		; AND ABORT TEST		
	054474	104410							TRAP	C\$ESCAPE
	054476	000724							.WORD	L10051-
8152										
8153	054500	004737	027566'		JSR	PC,CHKTXI		; TXI ?		
8154	054504	103010			BCC	140\$		; YES		
8155	054506				FTL					
	054506	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
8156	054512				ERRHRD	272.,ERR013,MSG003		; NO, REPORT ERROR		
	054512	104456							TRAP	C\$ERHRD
	054514	000420							.WORD	272
	054516	023530'							.WORD	ERR013
	054520	021736'							.WORD	MSG003
8157	054522				ESCAPE	TST		; AND ABORT TEST		
	054522	104410							TRAP	C\$ESCAPE
	054524	000676							.WORD	L10051
8158										
8159	054526	004737	030500'		JSR	PC,CLRTXI		; WRITE ONE TO CLEAR TXI		
8160								; ERROR ?		
8161	054532	103010			BCC	150\$		; NO		
8162	054534				FTL					
	054534	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		

```

8163 054540          ERRHRD  273.,ERR014,MSG003      ; YES, REPORT ERROR
      054540 104456
      054542 000421
      054544 023561'
      054546 021736'
8164 054550          ESCAPE  TST                    ; AND ABORT TEST
      054550 104410
      054552 000650
8165
8166          ;CHECK FIRST RING ENTRY
8167
8168 054554 012705 000620' 150$:  MOV    #TDRB,R5      ; CHECK TDRB OWNERSHIP
8169 054560 004737 027024'   JSR    PC,CHKOWN    ; OWN = PORT DRIVER ?
8170 054564 103010          BCC    160$        ; YES
8171 054566          FTL
      054566 004737 026652'   JSR    PC,CHKFTL    ; 'FATL' BI, SET?
8172 054572          ERRHRD  274.,ERR027            ; NO, REPORT ERROR
      054572 104456
      054574 000422
      054576 024540'
      054600 000000
8173 054602          ESCAPE  TST                    ; AND ABORT TEST
      054602 104410
      054604 000616
8174
8175 054606 012705 016324' 160$:  MOV    #TDR20A,R5   ; POINT TO EXPECTED TDRB
8176 054612 004737 032370'   JSR    PC,LDXTDR   ; LOAD INTO XTDRBO TABLE
8177 054616 012705 000620'   MOV    #TDRB,R5   ; CHECK TDRB
8178 054622 004737 027500'   JSR    PC,CHKTDR   ; ERRORS ?
8179 054626 103010          BCC    162$        ; NO
8180 054630          FTL
      054630 004737 026652'   JSR    PC,CHKFTL    ; 'FATL' BIT SET?
8181 054634          ERRHRD  275.,ERR033,MSG005      ; YES, REPORT ERROR
      054634 104456
      054636 000423
      054640 025166'
      054642 022042'
8182 054644          ESCAPE  TST                    ; AND ABORT TEST
      054644 104410
      054646 000554
8183
8184          ;CHECK SFCOND RING ENTRY
8185
8186 054650 012705 000630' 162$:  MOV    #TDRB+8.,R5  ; CHECK TDRB OWNERSHIP
8187 054654 004737 027024'   JSR    PC,CHKOWN    ; OWN = PORT DRIVER ?
8188 054660 103010          BCC    164$        ; YES
8189 054662          FTL
      054662 004737 026652'   JSR    PC,CHKFTL    ; 'FATL' BIT SET?
8190 054666          ERRHRD  276.,ERR028            ; NO, REPORT ERROR
      054666 104456
      TRAP  C$ERHRD
  
```



```

8216                ;CHECK FIRST RING ENTRY
8217
8218 055020 012705 000660' 190$:  MOV    #RDRB,R5      ; CHECK RDRB OWNERSHIP
8219 055024 004737 027024'      JSR    PC,CHKOWN    ; OWN = PORT DRIVER ?
8220 055030 103010                BCC    200$        ; YES
8221 055032                FTL
                                JSR    PC,CHKFTL      ; 'FATL' BIT SET?
                                ERRHRD 282.,ERR030    ; NO, REPORT ERROR
                                TRAP   C$ERHRD
                                .WORD 282
                                .WORD ERR030
                                .WORD 0
8222 055036                JSR    PC,CHKFTL
                                ERRHRD 282.,ERR030
                                TRAP   C$ERHRD
                                .WORD 282
                                .WORD ERR030
                                .WORD 0
                                055036 104456
                                055040 000432
                                055042 024753'
                                055044 000000
8223 055046                ESCAPE TST          ; AND ABORT TEST
                                TRAP   C$ESCAPE
                                .WORD L10051-.
                                055046 104410
                                055050 000352
8224                ;
8225 055052 012705 016434' 200$:  MOV    #RDR20A,R5    ; POINT TO EXPECTED RDRB
8226 055056 004737 032340'      JSR    PC,LDXRDR    ; LOAD INTO XRDRBO TABLE
8227 055062 012705 000660'      MOV    #RDRB,R5    ; CHECK RDRB
8228 055066 004737 027206'      JSR    PC,CHKRDR    ; ERRORS ?
8229 055072 103010                BCC    202$        ; NO
8230 055074                FTL
                                JSR    PC,CHKFTL      ; 'FATL' BIT SET?
                                ERRHRD 283.,ERR036,MSG006 ; YES, REPORT ERROR
                                TRAP   C$ERHRD
                                .WORD 283
                                .WORD ERR036
                                .WORD MSG006
8231 055100                JSR    PC,CHKFTL
                                ERRHRD 283.,ERR036,MSG006
                                TRAP   C$ERHRD
                                .WORD 283
                                .WORD ERR036
                                .WORD MSG006
                                055100 104456
                                055102 000433
                                055104 025345'
                                055106 022204'
8232 055110                ESCAPE TST          ; AND ABORT TEST
                                TRAP   C$ESCAPE
                                .WORD L10051-.
                                055110 104410
                                055112 000310
8233                ;CHECK SECOND RING ENTRY
8234
8235
8236 055114 012705 000670' 202$:  MOV    #RDRB+8.,R5    ; CHECK RDRB OWNERSHIP
8237 055120 004737 027024'      JSR    PC,CHKOWN    ; OWN = PORT DRIVER ?
8238 055124 103010                BCC    204$        ; YES
8239 055126                FTL
                                JSR    PC,CHKFTL      ; 'FATL' BIT SET?
                                ERRHRD 284.,ERR031    ; NO, REPORT ERROR
                                TRAP   C$ERHRD
                                .WORD 284
                                .WORD ERR031
                                .WORD 0
8240 055132                JSR    PC,CHKFTL
                                ERRHRD 284.,ERR031
                                TRAP   C$ERHRD
                                .WORD 284
                                .WORD ERR031
                                .WORD 0
                                055132 104456
                                055134 000434
                                055136 025060'
                                055140 000000
8241 055142                ESCAPE TST          ; AND ABORT TEST
                                TRAP   C$ESCAPE
                                .WORD L10051-.
                                055142 104410
                                055144 000256
8242                ;
8243 055146 012705 016444' 204$:  MOV    #RDR20B,R5    ; POINT TO EXPECTED RDRB
8244 055152 004737 032340'      JSR    PC,LDXRDR    ; LOAD INTO XRDRBO TABLE
8245 055156 012705 000670'      MOV    #RDRB+8.,R5 ; CHECK RDRB

```



```

8246 055162 004737 027206'      JSR    PC,CHKRDR      ; ERRORS ?
8247 055166 103010                BCC    210$          ; NO
8248 055170                FTL

      055170 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?
8249 055174                ERRHRD 285.,ERR037,MSG006 ; YES, REPORT ERROR
      055174 104456                TRAP  C$ERHRD
      055176 000435                .WORD 285
      055200 025433'                .WORD ERR037
      055202 022204'                .WORD MSG006
8250 055204                ESCAPE TST           ; AND ABORT TEST
      055204 104410                TRAP  C$ESCAPE
      055206 000214                .WORD L10051-.

8251
8252
8253 055210                ;
      210$:
8254 055210 012705 007054'      MOV    #RBUF2+14,R5  ; POINT TO CRC ADDRESS
8255 055214 022725 152120      CMP    #152120,(R5)+ ; 1ST CRC WORD BAD?
8256 055220 001003                BNE    222$          ; YES
8257 055222 022715 136614      CMP    #136614,(R5) ; 2ND CRC WORD BAD?
8258 055226 001422                BEQ    230$          ; NO, SKIP ERROR REPORT
8259 055230                ;
      222$:
8260 055230 012703 016574'      MOV    #XCRC,R3      ; POINT TO ERROR TABLE
8261 055234 012704 007054'      MOV    #RBUF2+14,R4  ; POINT TO ACTUAL CRC RECEIVED
8262 055240 012324                MOV    (R3)+,(R4)+   ; LOAD CRC ERROR TABLE
8263 055242 011314                MOV    (R3),(R4)     ;
8264 055244 012703 016574'      MOV    #XCRC,R3      ; POINT TO EXPECTED CRC TABLE
8265 055250 012723 152120      MOV    #152120,(R3)+ ; LOAD TABLE
8266 055254 012713 136614      MOV    #136614,(R3) ;
8267 055260                ERRHRD 286.,ERR023,MSG008 ; YES, REPORT ERROR
      055260 104456                TRAP  C$ERHRD
      055262 000436                .WORD 286
      055264 024417'                .WORD ERR023
      055266 022400'                .WORD MSG008
8268 055270                ESCAPE TST           ; AND ABORT TEST
      055270 104410                TRAP  C$ESCAPE
      055272 000130                .WORD L10051.

8269
      230$:
8270 055274                ;
8271 055274 012777 004100 122722 MOV    #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
8272 055302 112777 000117 122714 MOVB   #INTE!STOP,@PCSR0 ; ISSUE STOP PORT COMMAND
8273 055310 004737 026550'      JSR    PC,CHKDNI     ; DNI ?
8274 055314 103010                BCC    240$          ; YES
8275 055316                FTL

      055316 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?
8276 055322                ERRHRD 287.,ERR019,MSG003 ; NO, REPORT ERROR
      055322 104456                TRAP  C$ERHRD
      055324 000437                .WORD 287
      055326 024126'                .WORD ERR019
      055330 021736'                .WORD MSG003
8277 055332                ESCAPE TST           ; AND ABORT TEST
      055332 104410                TRAP  C$ESCAPE
      055334 000066                .WORD L10051-.

8278

```



8297  
 8298  
 8299  
 8300  
 8301  
 8302  
 8303  
 8304  
 8305  
 8306  
 8307  
 8308  
 8309  
 8310  
 8311  
 8312  
 8313  
 8314  
 8315  
 8316  
 8317  
 8318  
 8319  
 8320  
 8321  
 8322  
 8323  
 8324  
 8325  
 8326  
 8327  
 8328  
 8329

.SBTTL TEST 20: PHYSICAL ADDRESS TEST

```

*****
:
: THIS TEST VERIFIES THAT PHYSICAL ADDRESS DETECTION
: IS OPERATIONAL.
: A WRITE PHYSICAL ADDRESS FUNCTION IS USED TO SET
: THE DELUA'S PHYSICAL ADDRESS.
: INTERNAL LOOPBACKS ARE THEN PERFORMED WITH A
: CURRENTLY ENABLED AND THEN A CURRENTLY DISABLED
: DESTINATION ADDRESS.
: THE PHYSICAL ADDRESS IS THEN COMPLEMENTED AND THE
: TEST IS REPEATED.
:
: TEST SEQUENCE:
:   1. WRITE MODE REGISTER = INTERNAL LOOPBACK
:   2. WRITE RING FORMAT
:   3. WRITE PHYSICAL ADDRESS
:   4. SET UP RINGS AND BUFFERS
:      WITH DESTINATION ADDRESS = PHYSICAL ADDRESS
:   5. ISSUE START
:   6. CHECK FOR ERRORS
:   7. ISSUE STOP
:   8. SET UP RINGS AND BUFFERS
:      WITH DESTINATION ADDRESS NOT = PHYSICAL ADDRESS
:   9. ISSUE START
:  10. CHECK FOR NO RXI
:  11. ISSUE STOP
:  12. WRITE PHYSICAL ADDRESS WITH COMPLEMENTED VALUE
:  13. REPEAT STEPS 4 - 11
:
*****
  
```

8330 055424  
 055424  
 8331  
 8332 055424  
 055424 012704 061054'  
 055430 004737 032734'  
 8333 055434 004737 033434'  
 8334 055440 103034  
 8335 055442 012777 004100 122554  
 8336 055450 112777 000140 122546  
 8337 055456 004737 027736'  
 8338 055462 103010  
 8339 055464  
 055464 004737 026652'  
 8340 055470  
 055470 104456  
 055472 000443  
 055474 023030'

```

BGNTST
T20:
PNTMAC T20ID
MOV #T20ID,R4 ;GET POINTER TO TEST NAME MESSAGE
JSR PC,PNTID ;PRINT TEST NUMBER AND NAME
;
END OF MACRO EXPANSION OF 'PNTMAC'
JSR PC,TINIT ; IS A DEVICE RESET NEEDED?
BCC 30$ ; NO
MOV #DNI!INTE,@PCSRO ; ENABLE INTERRUPTS
MOVB #INTE!RSET,@PCSRO ; YES, RESET DELUA
JSR PC,CKDNI ; DNI ?
BCC 20$ ; YES
FTL
JSR PC,CHKFTL ; 'FATL' BIT SET?
ERRHRD 291.,ERR006,MSG003 ; NO, REPORT ERROR
TRAP C:ERHRD
.WORD 291
.WORD ERR006
  
```



```

8368
8369           ;WRITE MODE REGISTER = INTERNAL LOOPBACK
8370
8371 055636 012705 012600' 50$:  MOV  #WTRNGS,R5           ; DEFAULT WRITE MODE FUNCTION
8372 055642 004737 032002'      JSR  PC,LDPCBB           ; LOAD FUNCTION -> PCBB
8373 055646 013737 016472' 000302'  MOV  MODE21,PCBB+2     ; MODE = INTL LOOPBACK ONLY
8374 055654 012777 004100 122342  MOV  #DNI!INTE,@PCSR0  ; ENABLE INTERRUPTS
8375 055662 112777 000102 122334  MOVB #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
8376 055670 004737 026550'      JSR  PC,CHKDNI         ; DNI ?
8377 055674 103010              BCC  60$              ; YES
8378 055676
           JSR  PC,CHKFTL           ; 'FATL' BIT SET?
           ERRHRD 295.,ERR010,MSG003 ; NO, REPORT ERROR
           TRAP  C#ERHRD
           .WORD 295
           .WORD ERR010
           .WORD MSG003
8379 055702
           055702 104456
           055704 000447
           055706 023331'
           055710 021736'
8380 055712          ESCAPE TST           ; AND ABORT TEST
           TRAP  C#ESCAPE
           .WORD L10052-.
           055712 104410
           055714 003170
8381
8382 055716 004737 030264' 60$:  JSR  PC,CLRDNI         ; WRITE ONE TO CLEAR DNI
8383
8384 055722 103010              BCC  70$              ; ERROR ?
8385 055724
           FTL
           JSR  PC,CHKFTL           ; 'FATL' BIT SET?
           ERRHRD 296.,ERR006,MSG003 ; YES, REPORT ERROR
           TRAP  C#ERHRD
           .WORD 296
           .WORD ERR006
           .WORD MSG003
8386 055730
           055730 104456
           055732 000450
           055734 023030'
           055736 021736'
           ESCAPE TST           ; AND ABORT TEST
           TRAP  C#ESCAPE
           .WORD L10052-.
8387 055740
           055740 104410
           055742 003142
8388
8389           ;WRITE RING FORMAT
8390
8391 055744 012705 012540' 70$:  MOV  #WTRNGS,R5           ; DEFAULT WRITE RING FORMAT FUNCTION
8392 055750 004737 032002'      JSR  PC,LDPCBB           ; LOAD FUNCTION -> PCBB
8393 055754 012705 012704'      MOV  #RFRMT,R5         ; DEFAULT RING FORMAT
8394 055760 012700 000076        MOV  #6,R0             ; FORMAT = SIX WORDS
8395 055764 004737 032260'      JSR  PC,LDUDBB         ; LOAD RING FORMAT -> UDBB
8396 055770 012777 004100 122226  MOV  #DNI!INTE,@PCSR0  ; ENABLE INTERRUPTS
8397 055776 112777 000102 122220  MOVB #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
8398 056004 004737 026550'      JSR  PC,CHKDNI         ; DNI ?
8399 056010 103010              BCC  80$              ; YES
8400 056012
           JSR  PC,CHKFTL           ; 'FATL' BIT SET?
           ERRHRD 297.,ERR010,MSG003 ; NO, REPORT ERROR
           TRAP  C#ERHRD
           .WORD 297
8401 056016
           056016 104456
           056020 000451
    
```

	056022	023331'							.WORD	ERR010
	056024	021736'							.WORD	MSG003
8402	056026		ESCAPE	TST						
	056026	104410							TRAP	C\$ESCAPE
	056030	003054							.WORD	L10052-.
8403										
8404	056032	004737	030264'	80\$:	JSR	PC,CLRDNI				
8405										
8406	056036	103010			BCC	90\$				
8407	056040				FTL					
	056040	004737	026652'		JSR	PC,CHKFTL				
8408	056044				ERRHRD	300.,ERR006,MSG003				
	056044	104456							TRAP	C\$ERHRD
	056046	000454							.WORD	300
	056050	023030'							.WORD	ERR006
	056052	021736'							.WORD	MSG003
8409	056054		ESCAPE	TST						
	056054	104410							TRAP	C\$ESCAPE
	056056	003026							.WORD	L10052-.
8410										
8411										
8412										
8413	056060				90\$:					
8414	056060	012705	016060'		MOV	#ADR21,R5				
8415	056064	004737	032050'		JSR	PC,LDPHYA				
8416	056070	012705	012500'		MOV	#WTPHYA,R5				
8417	056074	004737	032002'		JSR	PC,LDPCBB				
8418	056100	012701	016060'		MOV	#ADR21,R1				
8419	056104	010102			MOV	R1,R2				
8420	056106	004737	033206'		JSR	PC,SRCDST				
8421	056112	012777	004100	122104	MOV	#DNI!INTE,@PCSR0				
8422	056120	112777	000102	122076	MOV	#INTE!GETCMD,@PCSR0				
8423	056126	004737	026550'		JSR	PC,CHKDNI				
8424	056132	103010			BCC	100\$				
8425	056134				FTL					
	056134	004737	026652'		JSR	PC,CHKFTL				
8426	056140				ERRHRD	301.,ERR010,MSG003				
	056140	104456							TRAP	C\$ERHRD
	056142	000455							.WORD	301
	056144	023331'							.WORD	ERR010
	056146	021736'							.WORD	MSG003
8427	056150		ESCAPE	TST						
	056150	104410							TRAP	C\$ESCAPE
	056152	002732							.WORD	L10052.
8428										
8429	056154	004737	030264'	100\$:	JSR	PC,CLRDNI				
8430										
8431	056160	103010			BCC	110\$				
8432	056162				FTL					
	056162	004737	026652'		JSR	PC,CHKFTL				
8433	056166				ERRHRD	302.,ERR006,MSG003				

```

056166 104456
056170 000456
056172 023030'
056174 021736'
8434 056176          ESCAPE TST          ; AND ABORT TEST
056176 104410
056200 002704          TRAP      C$ERHRD
                                .WORD      302
                                .WORD      ERR006
                                .WORD      MSG003
8435
8436          ;SET UP RINGS FOR ONE BUFFER LOOPBACK
8437
8438 056202 012705 014410' 110$: MOV    #TDRB1A,R5          ; DEFAULT ONE BUFFER TRANSMIT RING
8439 056206 004737 032164'      JSR    PC,LDTDRB          ; LOAD TDRB
8440 056212 012705 012750'      MOV    #RDRB1A,R5          ; DEFAULT ONE BUFFER RECEIVE RING
8441 056216 004737 032070'      JSR    PC,LDRDRB          ; LOAD RDRB
8442
8443          ;SET UP BUFFERS AND START
8444
8445 056222 005037 016562'      CLR    D0CRC              ; NO CRC
8446 056226 012737 000006 016560'  MOV    #6,BYTCNT          ; BYTES/PACKET
8447 056234 004737 033006'      JSR    PC,SETBUF          ; SET UP BUFFERS
8448 056240 012777 004100 121756  MOV    #DNI!INTE,@PCSR0    ; ENABLE INTERRUPTS
8449 056246 112777 000104 121750  MOVB   #INTE!START,@PCSR0  ; ISSUE START PORT COMMAND
8450 056254 004737 026550'      JSR    PC,CHKDNI          ; DNI?
8451 056260 103010          BCC    120$              ; YES
8452 056262          FTL
                                ; 'FATL' BIT SET?
                                JSR    PC,CHKFTL
                                ERRHRD 303.,ERR012,MSG003          ; NO, REPORT ERROR
8453 056266          TRAP      C$ERHRD
056266 104456          .WORD      303
056270 000457          .WORD      ERR012
056272 023447'          .WORD      MSG003
056274 021736'
8454 056276          ESCAPE TST          ; AND ABORT TEST
056276 104410
056300 002604          TRAP      C$ESCAPE
                                .WORD      L10052
8455
8456 056302 004737 030264' 120$: JSR    PC,CLRDN1          ; WRITE ONE TO CLEAR DNI
8457          BCC    130$          ; ERROR ?
8458 056306 103010          FTL                      ; NO
8459 056310
                                ; 'FATL' BIT SET?
                                JSR    PC,CHKFTL
                                ERRHRD 304.,ERR006,MSG003          ; YES, REPORT ERROR
8460 056314          TRAP      C$ERHRD
056314 104456          .WORD      304
056316 000460          .WORD      ERR006
056320 023030'          .WORD      MSG003
056322 021736'
8461 056324          ESCAPE TST          ; AND ABORT TEST
056324 104410
056326 002556          TRAP      C$ESCAPE
                                .WORD      L10052
8462
8463 056330 004737 027566' 130$: JSR    PC,CHKTXI          ; TXI ?
8464 056334 103010          BCC    140$          ; YES
8465 056336          FTL
    
```





8490	056474			ESCAPE	TST		; AND ABORT TEST		
	056474	104410						TRAP	C\$ESCAPE
	056476	002406						.WORD	L10052-
8491									
8492	056500	004737	027316'	i	JSR	PC,CHKRXI	; RXI ?		
8493	056504	103010		170\$:	BCC	180\$	; YES		
8494	056506				FTL				
	056506	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
8495	056512				ERRHRD	311.,ERR015,MSG003	; NO, REPORT ERROR		
	056512	104456						TRAP	C\$ERHRD
	056514	000467						.WORD	311
	056516	023627'						.WORD	ERR015
	056520	021736'						.WORD	MSG003
8496	056522				ESCAPE	TST	; AND ABORT TEST		
	056522	104410						TRAP	C\$ESCAPE
	056524	002360						.WORD	L10052-
8497									
8498	056526	004737	030432'	i	JSR	PC,CLRRXI	; WRITE ONE TO CLEAR RXI		
8499				180\$:			; ERROR ?		
8500	056532	103010			BCC	190\$	; NO		
8501	056534				FTL				
	056534	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
8502	056540				ERRHRD	312.,ERR016,MSG003	; YES, REPORT ERROR		
	056540	104456						TRAP	C\$ERHRD
	056542	000470						.WORD	312
	056544	023660'						.WORD	ERR016
	056546	021736'						.WORD	MSG003
8503	056550				ESCAPE	TST	; AND ABORT TEST		
	056550	104410						TRAP	C\$ESCAPE
	056552	002332						.WORD	L10052-
8504									
8505	056554	012705	000660'	i	MOV	@RDRB,R5	; CHECK RDRB OWNERSHIP		
8506	056560	004737	027024'	190\$:	JSR	PC,CHKOWN	; OWN = PORT DRIVER ?		
8507	056564	103010			BCC	200\$	; YES		
8508	056566				FTL				
	056566	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
8509	056572				ERRHRD	313.,ERR017	; NO, REPORT ERROR		
	056572	104456						TRAP	C\$ERHRD
	056574	000471						.WORD	313
	056576	023726'						.WORD	ERR017
	056600	000000						.WORD	0
8510	056602				ESCAPE	TST	; AND ABORT TEST		
	056602	104410						TRAP	C\$ESCAPE
	056604	002300						.WORD	L10052 .
8511									
8512	056606	012705	016454'	i	MOV	@RDR20C,R5	; POINT TO EXPECTED RDRB		
8513	056612	004737	032340'	200\$:	JSR	PC,LDXRDR	; LOAD INTO XRDRBO TABLE		
8514	056616	012705	000660'		MOV	@RDRB,R5	; CHECK RDRB		
8515	056622	004737	027206'		JSR	PC,CHKRDR	; ERRORS ?		
8516	056626	103010			BCC	210\$	; NO		
8517	056630				FTL				

```

056630 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
8518 056634      ERRHRD 314.,ERR021,MSG006 ; YES, REPORT ERROR      TRAP      C$ERHRD
056634 104456      .WORD 314
056636 000472      .WORD ERR021
056640 024267'      .WORD MSG006
056642 022204'
8519 056644      ESCAPE TST      ; AND ABORT TEST      TRAP      C$ESCAPE
056644 104410      .WORD L10052-.
056646 002236
8520      ;
8521      ;COMPARE RBUF WITH TBUF
8522
8523 056650 013705 016560' 210$: MOV      BYTCNT,R5      ; COMPARE DATA
8524 056654 004737 030712' JSR      PC,CMPDAT      ; DATA COMPARE ERROR ?
8525 056660 103006      BCC     220$          ; NO
8526 056662      ERRHRD 315.,ERR022,MSG007 ; YES, REPORT ERROR      TRAP      C$ERHRD
056662 104456      .WORD 315
056664 000473      .WORD ERR022
056666 024350'      .WORD MSG007
056670 022346'
8527 056672      ESCAPE TST      ; AND ABORT TEST      TRAP      C$ESCAPE
056672 104410      .WORD L10052 .
056674 002210
8528      ;
8529 056676 012705 006472' 220$: MOV      #RBUF+32,R5 ; POINT TO EXPECTED CRC
8530 056702 004737 030642' JSR      PC,CMPCRC      ; ERRORS ?
8531 056706 103006      BCC     230$          ; NO
8532 056710      ERRHRD 316.,ERR023,MSG008 ; YES, REPORT ERROR      TRAP      C$ERHRD
056710 104456      .WORD 316
056712 000474      .WORD ERR023
056714 024417'      .WORD MSG008
056716 022400'
8533 056720      ESCAPE TST      ; AND ABORT TEST      TRAP      C$ESCAPE
056720 104410      .WORD L10052 .
056722 022162
8534      ;
8535 056724      ;230$:
8536 056724 012777 004100 121272 MOV      #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
8537 056732 112777 000117 121264 MOVB     #INTE!STOP,@PCSR0 ; ISSUE STOP PORT COMMAND
8538 056740 004737 026550' JSR      PC,CHKDNI      ; DNI ?
8539 056744 103010      BCC     240$          ; YES
8540 056746      FTL
056746 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
8541 056752      ERRHRD 317.,ERR019,MSG003 ; NO, REPORT ERROR      TRAP      C$ERHRD
056752 104456      .WORD 317
056754 000475      .WORD ERR019
056756 024126'      .WORD MSG003
056760 021736'
8542 056762      ESCAPE TST      ; AND ABORT TEST      TRAP      C$ESCAPE
056762 104410      .WORD L10052-.
056764 002120
8543      ;
8544 056766 004737 030264' 240$: JSR      PC,CLRDMI      ; WRITE ONE TO CLEAR DNI
    
```

```

8545                                     ; ERROR ?
8546 056772 103010                       BCC 250$ ; NO
8547 056774                               FTL

      056774 004737 026652'             JSR PC,CHKFTL ; 'FATL' BIT SET?

8548 057000                               ERRHRD 320.,ERR006,MSG003 ; YES, REPORT ERROR
      057000 104456                               TRAP C$ERHRD
      057002 000500                               .WORD 320
      057004 023030'                               .WORD ERR006
      057006 021736'                               .WORD MSG003

8549 057010                               ESCAPE TST ; AND ABORT TEST
      057010 104410                               TRAP C$ESCAPE
      057012 002072                               .WORD L10052-.

8550                                     ;
8551                                     ;DESTINATION ADDRESS NOT = PHYSICAL ADDRESS
8552                                     ;
8553                                     ;SET UP RINGS FOR ONE BUFFER LOOPBACK
8554                                     ;
8555 057014                               250$:
8556 057014 004737 030236'             JSR PC,CLRCV ; CLEAR RECEIVE BUFFER
8557 057020 012705 014410'             MOV #TDRB1A,R5 ; DEFAULT ONE BUFFER TRANSMIT RING
8558 057024 004737 032164'             JSR PC,LDTDRB ; LOAD TDRB
8559 057030 012705 012750'             MOV #RDRB1A,R5 ; DEFAULT ONE BUFFER RECEIVE RING
8560 057034 004737 032070'             JSR PC,LDRDRB ; LOAD RDRB

8561                                     ;
8562                                     ;SET UP BUFFERS AND START
8563                                     ;
8564 057040 012701 016060'             MOV #ADR21,R1 ; SET SOURCE = PHYSICAL ADDRESS
8565 057044 012702 016066'             MOV #ADR21C,R2 ; DEST = COMPLEMENTED ADDRESS
8566 057050 004737 033206'             JSR PC,SRCDST ; LOAD PACKET ADDRESSES
8567 057054 005037 016562'             CLR DDCRC ; NO CRC
8568 057060 012737 000006 016560'     MOV #6,BYTCNT ; BYTES/PACKET
8569 057066 004737 033006'             JSR PC,SETBUF ; SET UP BUFFERS
8570 057072 012777 004100 121124     MOV #DNI!INTE,@PCSRO ; ENABLE INTERRUPTS
8571 057100 112777 000104 121116     MOVB #INTE!START,@PCSRO ; ISSUE START PORT COMMAND
8572 057106 004737 026550'             JSR PC,CHKDNI ; DNI?
8573 057112 103010                       BCC 260$ ; YES
8574 057114                               FTL

      057114 004737 026652'             JSR PC,CHKFTL ; 'FATL' BIT SET?

8575 057120                               ERRHRD 321.,ERR012,MSG003 ; NO, REPORT ERROR
      057120 104456                               TRAP C$ERHRD
      057122 000501                               .WORD 321
      057124 023447'                               .WORD ERR012
      057126 021736'                               .WORD MSG003

8576 057130                               ESCAPE TST ; AND ABORT TEST
      057130 104410                               TRAP C$ESCAPE
      057132 001752                               .WORD L10052-.

8577                                     ;
8578 057134 004737 030264'             260$: JSR PC,CLRDN ; WRITE ONE TO CLEAR DNI
8579                                     ; ERROR ?
8580 057140 103010                       BCC 270$ ; NO
8581 057142                               FTL

      057142 004737 026652'             JSR PC,CHKFTL ; 'FATL' BIT SET?
    
```

8582	057146			ERRHRD	322.,ERR006,MSG003	; YES, REPORT ERROR		
	057146	104456					TRAP	C#ERHRD
	057150	000502					.WORD	322
	057152	023030'					.WORD	ERR006
	057154	021736'					.WORD	MSG003
8583	057156			ESCAPE	TST	; AND ABORT TEST		
	057156	104410					TRAP	C#ESCAPE
	057160	001724					.WORD	L10052-.
8584								
8585	057162	004737	027566'	i	JSR	PC,CHKTXI		; TXI ?
8586	057166	103010		270\$:	BCC	280\$		; YES
8587	057170				FTL			
	057170	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
8588	057174				ERRHRD	323.,ERR013,MSG003		; NO, REPORT ERROR
	057174	104456					TRAP	C#ERHRD
	057176	000503					.WORD	323
	057200	023530'					.WORD	ERR013
	057202	021736'					.WORD	MSG003
8589	057204				ESCAPE	TST		; AND ABORT TEST
	057204	104410					TRAP	C#ESCAPE
	057206	001676					.WORD	L10052-.
8590								
8591	057210	004737	030500'	i	JSR	PC,CLRXTXI		; WRITE ONE TO CLEAR TXI
8592				280\$:				; ERROR ?
8593	057214	103010			BCC	290\$		; NO
8594	057216				FTL			
	057216	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
8595	057222				ERRHRD	324.,ERR014,MSG003		; YES, REPORT ERROR
	057222	104456					TRAP	C#ERHRD
	057224	000504					.WORD	324
	057226	023561'					.WORD	ERR014
	057230	021736'					.WORD	MSG003
8596	057232				ESCAPE	TST		; AND ABORT TEST
	057232	104410					TRAP	C#ESCAPE
	057234	001650					.WORD	L10052 .
8597								
8598	057236	012705	000620'	i	MOV	#TDRB,R5		; CHECK TDRB OWNERSHIP
8599	057242	004737	027024'	290\$:	JSR	PC,CHKOWN		; OWN = PORT DRIVER ?
8600	057246	103010			BCC	300\$		; YES
8601	057250				FTL			
	057250	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
8602	057254				ERRHRD	325.,ERR018		; NO, REPORT ERROR
	057254	104456					TRAP	C#ERHRD
	057256	000505					.WORD	325
	057260	024026'					.WORD	ERR018
	057262	000000					.WORD	0
8603	057264				ESCAPE	TST		; AND ABORT TEST
	057264	104410					TRAP	C#ESCAPE
	057266	001616					.WORD	L10052 .
8604				i				

```

8605 057270 012705 016344'      300$:  MOV    #TDR21X,R5      ; POINT TO EXPECTED TDRB
8606 057274 004737 032370'      JSR    PC,LDXTDR      ; LOAD INTO XTDRBO TABLE
8607 057300 012705 000620'      MOV    #TDRB,R5      ; CHECK TDRB
8608 057304 004737 027500'      JSR    PC,CHKTDR      ; ERRORS ?
8609 057310 103010                BCC    310$           ; NO
8610 057312                FTL

      057312 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?

8611 057316                ERRHRD 326.,ERRG20,MSG005 ; YES, REPORT ERROR
      057316 104456                TRAP  C$ERHRD
      057320 000506                .WORD 326
      057322 024206'                .WORD ERR020
      057324 022042'                .WORD MSG005

8612 057326                ESCAPE TST            ; AND ABORT TEST
      057326 104410                TRAP  C$ESCAPE
      057330 001554                .WORD L10052-.

8613                ;
8614 057332 004737 032422'      310$:  JSR    PC,NORXI      ; RXI ?
8615 057336 103010                BCC    320$           ; NO
8616 057340                FTL

      057340 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?

8617 057344                ERRHRD 327.,ERR039    ; YES, REPORT ERROR
      057344 104456                TRAP  C$ERHRD
      057346 000507                .WORD 327
      057350 025575'                .WORD ERR039
      057352 000000                .WORD 0

8618 057354                ESCAPE TST            ; AND ABORT TEST
      057354 104410                TRAP  C$ESCAPE
      057356 001526                .WORD L10052-.

8619                ;
8620 057360                320$:  MOV    #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
8621 057360 012777 004100 120636  MOVB   #INTE!STOP,@PCSR0 ; ISSUE STOP PORT COMMAND
8622 057366 112777 000117 120630  JSR    PC,CHKDNI      ; DNI ?
8623 057374 004737 026550'      BCC    330$           ; YES
8624 057400 103010                FTL

      057402 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?

8626 057406                ERRHRD 330.,ERR019,MSG003 ; NO, REPORT ERROR
      057406 104456                TRAP  C$ERHRD
      057410 000512                .WORD 330
      057412 024126'                .WORD ERR019
      057414 021736'                .WORD MSG003

8627 057416                ESCAPE TST            ; AND ABORT TEST
      057416 104410                TRAP  C$ESCAPE
      057420 001464                .WORD L10052-.

8628                ;
8629 057422 004737 030264'      330$:  JSR    PC,CLRDN1    ; WRITE ONE TO CLEAR DNI
8630                ; ERROR ?
8631 057426 103010                BCC    340$           ; NO
8632 057430                FTL

      057430 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?
  
```

```

8633 057434          ERRHRD  331.,ERR006,MSG003      ; YES, REPORT ERROR
      057434 104456
      057436 000513
      057440 023030'
      057442 021736'
8634 057444          ESCAPE  TST                    ; AND ABORT TEST
      057444 104410
      057446 001436
8635
8636                ;REPEAT WITH COMPLEMENTED PHYSICAL ADDRESS
8637
8638                ;WRITE PHYSICAL ADDRESS
8639
8640 057450          340$:
8641 057450 004737 030236'      JSR      PC,CLRCV          ; CLEAR RECEIVE BUFFER
8642 057454 012705 016066'      MOV      @ADR21C,R5       ; GET NEW PHYSICAL ADDRESS
8643 057460 004737 032050'      JSR      PC,LDPHYA       ; SAVE IT IN DEFAULT TABLE
8644 057464 012705 012500'      MOV      @WTPHYA,R5     ; DEFAULT WRITE PHYSICAL ADDR FUNC
8645 057470 004737 032002'      JSR      PC,LDPCBB      ; LOAD FUNCTION -> PCBB
8646 057474 012701 016066'      MOV      @ADR21C,R1     ; GET NEW SOURCE ADDRESS
8647 057500 010102          MOV      R1,R2           ; SOURCE = DESTINATION
8648 057502 004737 033206'      JSR      PC,SRCDST      ; SAVE PACKET ADDRESSES
8649 057506 012777 004100 120510  MOV      @DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
8650 057514 112777 000102 120502  MOVB    @INTE!GETCMD,@P 3R0 ; ISSUE GET_CMD PORT COMMAND
8651 057522 004737 026550'      JSR      PC,CHKDNI      ; DNI ?
8652 057526 103010          BCC      350$           ; YES
8653 057530          FTL
      057530 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
8654 057534          ERRHRD  332.,ERR010,MSG003      ; NO, REPORT ERROR
      057534 104456
      057536 000514
      057540 023331'
      057542 021736'
8655 057544          ESCAPE  TST                    ; AND ABORT TEST
      057544 104410
      057546 001336
8656
8657 057550 004737 030264'      350$: JSR      PC,CLR0NI      ; WRITE ONE TO CLEAR DNI
8658
8659 057554 103010          BCC      360$           ; ERROR ?
8660 057556          FTL
      057556 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
8661 057562          ERRHRD  333.,ERR006,MSG003      ; YES, REPORT ERROR
      057562 104456
      057564 000515
      057566 023030'
      057570 021736'
8662 057572          ESCAPE  TST                    ; AND ABORT TEST
      057572 104410
      057574 001310
8663
8664                ;SET UP RINGS FOR ONE BUFFER LOOPBACK
    
```

```

8665
8666 057576 012705 014410' 360$: MOV #TDRB1A,R5 ; DEFAULT ONE BUFFER TRANSMIT RING
8667 057602 004737 032164' JSR PC,LDTDRB ; LOAD TDRB
8668 057606 012705 012750' MOV #PDRB1A,R5 ; DEFAULT ONE BUFFER RECEIVE RING
8669 057612 004737 032070' JSR PC,LDRDRB ; LOAD RDRB
8670
8671 ;SET UP BUFFERS AND START
8672
8673 057616 005037 016562' CLR DDCRC ; NO CRC
8674 057622 012737 000006 016560' MOV #6,BYTCNT ; BYTES/PACKET
8675 057630 004737 033006' JSR PC,SETBUF ; SET UP BUFFERS
8676 057634 012777 004100 120362 MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
8677 057642 112777 000104 120354 MOVB #INTE!START,@PCSR0 ; ISSUE START PORT COMMAND
8678 057650 004737 026550' JSR PC,CHKDNI ; DNI?
8679 057654 103010 BCC 370$ ; YES
8680 057656 004737 026652' FTL
      JSR PC,CHKFTL ; 'FATL' BIT SET?
8681 057662 ERRHRD 334.,ERR012,MSG003 ; NO, REPORT ERROR
      057662 104456 TRAP C$ERHRD
      057664 000516 .WORD 334
      057666 023447' .WORD ERR012
      057670 021736' .WORD MSG003
8682 057672 ESCAPE TST ; AND ABORT TEST
      057672 104410 TRAP C$ESCAPE
      057674 001210 .WORD L10052-.
8683
8684 057676 004737 030264' 370$: JSR PC,CLR DNI ; WRITE ONE TO CLEAR DNI
8685 BCC 380$ ; ERROR ?
8686 057702 103010 FTL ; NO
8687 057704 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?
      ERRHRD 335.,ERR006,MSG003 ; YES, REPORT ERROR
      057710 104456 TRAP C$ERHRD
      057712 000517 .WORD 335
      057714 023030' .WORD ERR006
      057716 021736' .WORD MSG003
8689 057720 ESCAPE TST ; AND ABORT TEST
      057720 104410 TRAP C$ESCAPE
      057722 001162 .WORD L10052 .
8690
8691 057724 004737 027566' 380$: JSR PC,CHKTXI ; TXI ?
8692 057730 103010 BCC 390$ ; YES
8693 057732 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?
      ERRHRD 336.,ERR013,MSG003 ; NO, REPORT ERROR
      057736 104456 TRAP C$ERHRD
      057740 000520 .WORD 336
      057742 023530' .WORD ERR013
      057744 021736' .WORD MSG003
8695 057746 ESCAPE TST ; AND ABORT TEST
      057746 104410 TRAP C$ESCAPE
  
```

	057750	001134						.WORD	L10052-.
8696									
8697	057752	004737	030500'	i	390\$:	JSR	PC,CLRTXI		; WRITE ONE TO CLEAR TXI
8698									; ERROR ?
8699	057756	103010				BCC	400\$		; NO
8700	057760					FTL			
	057760	004737	026652'			JSR	PC,CHKFTL		; 'FATL' BIT SET?
8701	057764					ERRHRD	337.,ERR014,MSG003		; YES, REPORT ERROR
	057764	104456						TRAP	C\$ERHRD
	057766	000521						.WORD	337
	057770	023561'						.WORD	ERR014
	057772	021736'						.WORD	MSG003
8702	057774					ESCAPE	TST		; AND ABORT TEST
	057774	104410						TRAP	C\$ESCAPE
	057776	001106						.WORD	L10052-.
8703									
8704	060000	012705	000620'	i	400\$:	MOV	#TDRB,R5		; CHECK TDRB OWNERSHIP
8705	060004	004737	027024'			JSR	PC,CHKOWN		; OWN = PORT DRIVER ?
8706	060010	103010				BCC	410\$		; YES
8707	060012					FTL			
	060012	004737	026652'			JSR	PC,CHKFTL		; 'FATL' BIT SET?
8708	060016					ERRHRD	340.,ERR018		; NO, REPORT ERROR
	060016	104456						TRAP	C\$ERHRD
	060020	000524						.WORD	340
	060022	024026'						.WORD	ERR018
	060024	000000						.WORD	0
8709	060026					ESCAPE	TST		; AND ABORT TEST
	060026	104410						TRAP	C\$ESCAPE
	060030	001054						.WORD	L10052-.
8710									
8711	060032	012705	016264'	i	410\$:	MOV	#TDR14A,R5		; POINT TO EXPECTED TDRB
8712	060036	004737	032370'			JSR	PC,LDXTDR		; LOAD INTO XTDRBO TABLE
8713	060042	012705	000620'			MOV	#TDRB,R5		; CHECK TDRB
8714	060046	004737	027500'			JSR	PC,CHKTDR		; ERRORS ?
8715	060052	103010				BCC	420\$		; NO
8716	060054					FTL			
	060054	004737	026652'			JSR	PC,CHKFTL		; 'FATL' BIT SET?
8717	060060					ERRHRD	341.,ERR020,MSG005		; YES, REPORT ERROR
	060060	104456						TRAP	C\$ERHRD
	060062	000525						.WORD	341
	060064	024206'						.WORD	ERR020
	060066	022042'						.WORD	MSG005
8718	060070					ESCAPE	TST		; AND ABORT TEST
	060070	104410						TRAP	C\$ESCAPE
	060072	001012						.WORD	L10052-.
8719									
8720	060074	004737	027316'	i	420\$:	JSR	PC,CHKRXI		; RXI ?
8721	060100	103010				BCC	430\$		; YES
8722	060102					FTL			
	060102	004737	026652'			JSR	PC,CHKFTL		; 'FATL' BIT SET?



```

8723 060106          ERRHRD  342.,ERR015,MSG003      ; NO, REPORT ERROR
      060106 104456
      060110 000526
      060112 023627'
      060114 021736'
8724 060116          ESCAPE  TST                  ; AND ABORT TEST
      060116 104410
      060120 000764
8725
8726 060122 004737 030432'      ; 430$: JSR      PC,CLRRXI      ; WRITE ONE TO CLEAR RXI
8727
8728 060126 103010          BCC      440$                ; ERROR ?
8729 060130          FTL
      060130 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
8730 060134          ERRHRD  343.,ERR016,MSG003      ; YES, REPORT ERROR
      060134 104456
      060136 000527
      060140 023660'
      060142 021736'
8731 060144          ESCAPE  TST                  ; AND ABORT TEST
      060144 104410
      060146 000736
8732
8733 060150 012705 000660'      ; 440$: MOV      @RDRB,R5      ; CHECK RDRB OWNERSHIP
8734 060154 004737 027024'      JSR      PC,CHKOWN        ; OWN = PORT DRIVER ?
8735 060160 103010          BCC      450$                ; YES
8736 060162          FTL
      060162 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
8737 060166          ERRHRD  344.,ERR017              ; NO, REPORT ERROR
      060166 104456
      060170 000530
      060172 023726'
      060174 000000
8738 060176          ESCAPE  TST                  ; AND ABORT TEST
      060176 104410
      060200 000704
8739
8740 060202 012705 016454'      ; 450$: MOV      @RDR20C,R5   ; POINT TO EXPECTED RDRB
8741 060206 004737 032340'      JSR      PC,LDXRDR        ; LOAD INTO XRDRBO TABLE
8742 060212 012705 000660'      MOV      @RDRB,R5        ; CHECK RDRB
8743 060216 004737 027206'      JSR      PC,CHKRDR        ; ERRORS ?
8744 060222 103010          BCC      460$                ; NO
8745 060224          FTL
      060224 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
8746 060230          ERRHRD  345.,ERR021,MSG006      ; YES, REPORT ERROR
      060230 104456
      060232 000531
      060234 024267'
      060236 022204'
8747 060240          ESCAPE  TST                  ; AND ABORT TEST
  
```

TRAP C#ERHRD  
 .WORD 342  
 .WORD ERR015  
 .WORD MSG003

TRAP C#ESCAPE  
 .WORD L10052-.

TRAP C#ERHRD  
 .WORD 343  
 .WORD ERR016  
 .WORD MSG003

TRAP C#ESCAPE  
 .WORD L10052-.

TRAP C#ERHRD  
 .WORD 344  
 .WORD ERR017  
 .WORD 0

TRAP C#ESCAPE  
 .WORD L10052-.

TRAP C#ERHRD  
 .WORD 345  
 .WORD ERR021  
 .WORD MSG006

	060240	104410					TRAP	C#ESCAPE
	060242	000642					.WORD	L10052-.
8748								
8749								
8750								
8751	060244	013705	016560'	460#:	MOV	BYTCNT,R5		; COMPARE DATA
8752	060250	004737	030712'		JSR	PC,CMPDAT		; DATA COMPARE ERROR ?
8753	060254	103006			BCC	470#		; NO
8754	060256				ERRHRD	346.,ERR022,MSG007		; YES, REPORT ERROR
	060256	104456					TRAP	C#ERHRD
	060260	000532					.WORD	346
	060262	024350'					.WORD	ERR022
	060264	022346'					.WORD	MSG007
8755	060266				ESCAPE	TST		; AND ABORT TEST
	060266	104410					TRAP	C#ESCAPE
	060270	000614					.WORD	L10052-.
8756								
8757	060272				470#:			
8758	060272	012705	006472'		MOV	#RBUF+26.,R5		; CHECK CRC
8759	060276	004737	030642'		JSR	PC,CMPCRC		; ERRORS ?
8760	060302	103006			BCC	480#		; NO
8761	060304				ERRHRD	347.,ERR023,MSG008		; YES, REPORT ERROR
	060304	104456					TRAP	C#ERHRD
	060306	000533					.WORD	347
	060310	024417'					.WORD	ERR023
	060312	022400'					.WORD	MSG008
8762	060314				ESCAPE	TST		; AND ABORT TEST
	060314	104410					TRAP	C#ESCAPE
	060316	000566					.WORD	L10052-.
8763								
8764	060320				480#:			
8765	060320	012777	004100 117676		MOV	#DNI!INTE,#PCSR0		; ENABLE INTERRUPTS
8766	060326	112777	000117 117670		MOVB	#INTE!STOP,#PCSR0		; ISSUE STOP PORT COMMAND
8767	060334	004737	026550'		JSR	PC,CHKDNI		; DNI ?
8768	060340	103010			BCC	490#		; YES
8769	060342				FTL			
	060342	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
8770	060346				ERRHRD	350.,ERR019,MSG003		; NO, REPORT ERROR
	060346	104456					TRAP	C#ERHRD
	060350	000536					.WORD	350
	060352	024126'					.WORD	ERR019
	060354	021736'					.WORD	MSG003
8771	060356				ESCAPE	TST		; AND ABORT TEST
	060356	104410					TRAP	C#ESCAPE
	060360	000524					.WORD	L10052-.
8772								
8773	060362	004737	030264'		490#:			
8774					JSR	PC,CLRDNI		; WRITE ONE TO CLEAR DNI
8775	060366	103010			BCC	500#		; ERROR ?
8776	060370				FTL			; NO
	060370	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
8777	060374				ERRHRD	351.,ERR006,MSG003		; YES, REPORT ERROR
	060374	104456					TRAP	C#ERHRD

060376	000537					.WORD	351
060400	023030'					.WORD	ERR006
050402	021736'					.WORD	MSG003
8778	060404			ESCAPE TST			; AND ABORT TEST
	060404	104410				TRAP	C\$ESCAPE
	060406	000476				.WORD	L10052-
8779							
8780							
8781							
8782							
8783							
8784	060410			500\$:			
8785	060410	004737	030236'	JSR	PC,CLRCV		; CLEAR RECEIVE BUFFER
8786	060414	012705	014410'	MOV	\$TDRB1A,R5		; DEFAULT ONE BUFFER TRANSMIT RING
8787	060420	004737	032164'	JSR	PC,LDTDRB		; LOAD TDRB
8788	060424	012705	012750'	MOV	\$RDRB1A,R5		; DEFAULT ONE BUFFER RECEIVE RING
8789	060430	004737	032070'	JSR	PC,LDRDRB		; LOAD RDRB
8790							
8791							
8792							
8793	060434	012701	016066'	MOV	\$ADR21C,R1		; GET PHYSICAL ADDRESS FOR SOURCE
8794	060440	012702	016060'	MOV	\$ADR21,R2		; COMPLIMENT = DESTINATION ADDRESS
8795	060444	004737	033206'	JSR	PC,SRCDST		; SAVE FOR PACKET ASSEMBLY
8796	060450	005037	016562'	CLR	DOCRC		; NO CRC
8797	060454	012737	000006 016560'	MOV	\$6,BYTCNT		; BYTES/PACKET
8798	060462	004737	033006'	JSR	PC,SETBUF		; SET UP BUFFERS
8799	060466	012777	004100 117530	MOV	\$DNI!INTE,@PCSR0		; ENABLE INTERRUPTS
8800	060474	112777	000104 117522	MOV	\$INTE!START,@PCSR0		; ISSUE START PORT COMMAND
8801	060502	004737	026550'	JSR	PC,CHKDNI		; DNI?
8802	060506	103010		BCC	510\$		; YES
8803	060510			FTL			
	060510	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?
8804	060514			ERRHRD	352.,ERR012,MSG003		; NO, REPORT ERROR
	060514	104456				TRAP	C\$ERHRD
	060516	000540				.WORD	352
	060520	023447'				.WORD	ERR012
	060522	021736'				.WORD	MSG003
8805	060524			ESCAPE TST			; AND ABORT TEST
	060524	104410				TRAP	C\$ESCAPE
	060526	000356				.WORD	L10052 .
8806							
8807	060530	004737	030264'	510\$:	JSR	PC,CLRDN1	; WRITE ONE TO CLEAR DNI
8808							; ERROR ?
8809	060534	103010		BCC	520\$		; NO
8810	060536			FTL			
	060536	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?
8811	060542			ERRHRD	353.,ERR006,MSG003		; YES, REPORT ERROR
	060542	104456				TRAP	C\$ERHRD
	060544	000541				.WORD	353
	060546	023030'				.WORD	ERR006
	060550	021736'				.WORD	MSG003
8812	060552			ESCAPE TST			; AND ABORT TEST
	060552	104410				TRAP	C\$ESCAPE

	060554	000330					.WORD	L10052
8813				i				
8814	060556	004737	027566'	520\$:	JSR	PC,CHKTXI		; TXI ?
8815	060562	103010			BCC	530\$		; YES
8816	060564				FTL			
	060564	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
8817	060570				ERRHRD	354.,ERR013,MSG003		; NO. REPORT ERROR
	060570	104456					TRAP	C\$ERHRD
	060572	000542					.WORD	354
	060574	023530'					.WORD	ERR013
	060576	021736'					.WORD	MSG003
8818	060600				ESCAPE	TST		; AND ABORT TEST
	060600	104410					TRAP	C\$ESCAPE
	060602	000302					.WORD	L10052-.
8819				i				
8820	060604	004737	030500'	530\$:	JSR	PC,CLRTXI		; WRITE ONE TO CLEAR TXI
8821								; ERROR ?
8822	060610	103010			BCC	540\$		; NO
8823	060612				FTL			
	060612	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
8824	060616				ERRHRD	355.,ERR014,MSG003		; YES. REPORT ERROR
	060616	104456					TRAP	C\$ERHRD
	06.620	000543					.WORD	355
	060622	023561'					.WORD	ERR014
	060624	021736'					.WORD	MSG003
8825	060626				ESCAPE	TST		; AND ABORT TEST
	060626	104410					TRAP	C\$ESCAPE
	060630	000254					.WORD	L10052-.
8826				i				
8827	060632	012705	000620'	540\$:	MOV	#TDRB,R5		; CHECK TDRB OWNERSHIP
8828	060636	004737	027024'		JSR	PC,CHKOWN		; OWN = PORT DRIVER ?
8829	060642	103010			BCC	550\$		; YES
8830	060644				FTL			
	060644	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
8831	060650				ERRHRD	356.,ERR018		; NO. REPORT ERROR
	060650	104456					TRAP	C\$ERHRD
	060652	000544					.WORD	356
	060654	024026'					.WORD	ERR018
	060656	000000					.WORD	0
8832	060660				ESCAPE	TST		; AND ABORT TEST
	060660	104410					TRAP	C\$ESCAPE
	060662	000222					.WORD	L10052-.
8833				i				
8834	060664	012705	016344'	550\$:	MOV	#TDR21X,R5		; POINT TO EXPECTED TDRB
8835	060670	004737	032370'		JSR	PC,LDXTDR		; LOAD INTO XTDRB0 TABLE
8836	060674	012705	000620'		MOV	#TDRB,R5		; CHECK TDRB
8837	060700	004737	027500'		JSR	PC,CHKTDR		; ERRORS ?
8838	060704	103010			BCC	560\$		; NO
8839	060706				FTL			
	060706	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?

8840	060712				ERRHRD	357.,ERR020,MSG005		; YES, REPORT ERROR		
	050712	104456							TRAP	C\$ERHRD
	060714	000545							.WORD	357
	060716	024206'							.WORD	ERR020
	060720	022042'							.WORD	MSG005
8841	060722				ESCAPE	TST		; AND ABORT TEST		
	060722	104410							TRAP	C\$ESCAPE
	060724	000160							.WORD	L10052-.
8842										
8843	060726	004737	032422'		JSR	PC,NORXI		; RXI ?		
8844	060732	103010			BCC	570\$		; NO		
8845	060734				FTL					
	060734	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
8846	060740				ERRHRD	360.,ERR039		; YES, REPORT ERROR		
	060740	104456							TRAP	C\$ERHRD
	060742	000550							.WORD	360
	060744	025575'							.WORD	ERR039
	060746	000000							.WORD	0
8847	060750				ESCAPE	TST		; AND ABORT TEST		
	060750	104410							TRAP	C\$ESCAPE
	060752	000132							.WORD	L10052-.
8848										
8849	060754									
8850	060754	012777	004100	117242	MOV	#DNI!INTE,@PCSR0		; ENABLE INTERRUPTS		
8851	060762	112777	000117	117234	MOVB	#INTE!STOP,@PCSR0		; ISSUE STOP PORT COMMAND		
8852	060770	004737	026550'		JSR	PC,CHKDNI		; DNI ?		
8853	060774	103010			BCC	580\$		; YES		
8854	060776				FTL					
	060776	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
8855	061002				ERRHRD	361.,ERR019,MSG003		; NO, REPORT ERROR		
	061002	104456							TRAP	C\$ERHRD
	061004	000551							.WORD	361
	061006	024126'							.WORD	ERR019
	061010	021736'							.WORD	MSG003
8856	061012				ESCAPE	TST		; AND ABORT TEST		
	061012	104410							TRAP	C\$ESCAPE
	061014	000070							.WORD	L10052 .
8857										
8858	061016	004737	030264'		JSR	PC,CLR DNI		; WRITE ONE TO CLEAR DNI		
8859								; ERROR ?		
8860	061022	103010			BCC	590\$		; NO		
8861	061024				FTL					
	061024	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
8862	061030				ERRHRD	362.,ERR006,MSG003		; YES, REPORT ERROR		
	061030	104456							TRAP	C\$ERHRD
	061032	000552							.WORD	362
	061034	023030'							.WORD	ERR006
	061036	021736'							.WORD	MSG003
8863	061040				ESCAPE	TST		; AND ABORT TEST		
	061040	104410							TRAP	C\$ESCAPE

```
061042 000042 .WORD L10052-.
8864 061044
8865 061044 004737 030346' 5904: JSR PC,CLINTR ; INSURE DELUA INTR BITS CLEAR
8866
8867 061050 EXIT TST
061050 104432 TRAP C$EXIT
061052 000032 .WORD L10052-.
8868
8869 ;LOCAL TEST MESSAGE
8870
8871 061054 104 105 114 T20ID:.ASCIZ 'DELUA PHYSICAL ADDRESS '
061057 125 101 040
061062 120 110 131
061065 123 111 103
061070 101 114 040
061073 101 104 104
061076 122 105 123
061101 123 040 000
8872 .EVEN
8873
8874 061104 ENDTST
061104
061104 104401 L10052: TRAP C$ETST
```

8876  
8877  
8878  
8879  
8880  
8881  
8882  
8883  
8884  
8885  
8886  
8887  
8888  
8889  
8890  
8891  
8892  
8893  
8894  
8895  
8896  
8897  
8898  
8899  
8900  
8901  
8902  
8903  
8904  
8905  
8906  
8907  
8908  
8909  
8910  
8911

.SBTTL TEST 21: MULTICAST ADDRESS TEST

```

*****
:
: THIS TEST VERIFIES THAT MULTICAST ADDRESSING
: IS OPERATIONAL.
: A WRITE MULTICAST ADDRESS LIST FUNCTION IS USED TO SET
: THE DELUA'S MULTICAST ADDRESS LIST.
: INTERNAL LOOPBACKS ARE THEN PERFORMED WITH
: CURRENTLY ENABLED AND THEN CURRENTLY DISABLED
: MULTICAST DESTINATION ADDRESSES.
: THE MULTICAST ADDRESS LIST IS THEN COMPLEMENTED AND THE
: TEST IS REPEATED.
:
: TEST SEQUENCE:
: 1. WRITE MODE REGISTER = INTERNAL LOOPBACK
: 2. WRITE RING FORMAT
: 3. WRITE PHYSICAL ADDRESS
: 4. WRITE MULTICAST ADDRESS LIST
: 5. SET UP RINGS AND BUFFERS
: WITH DESTINATION ADDRESS = MULTICAST ADDRESS
: 6. ISSUE START
: 7. CHECK FOR ERRORS
: 8. ISSUE STOP
: 9. REPEAT STEPS 5 - 8 FOR ALL TEN LIST ENTRIES
: 10. SET UP RINGS AND BUFFERS
: WITH DESTINATION ADDRESS = COMPLEMENTED MULTICAST ADDRESS
: 11. ISSUE START
: 12. CHECK FOR NO RXI
: 13. ISSUE STOP
: 14. REPEAT STEPS 10 - 13 FOR ALL TEN ENTRIES
: 15. WRITE MULTICAST ADDRESS LIST WITH COMPLEMENTED VAULES
: 16. REPEAT STEPS 5 - 14
:
*****
    
```

8912 061106  
061106  
8913  
8914 061106

BGNTST

T21::

PNTMAC T21ID

061106 012704 065006'  
061112 004737 032734'

```

MOV    #T21ID,R4      ;GET POINTER TO TEST NAME MESSAGE
JSR    PC,PNTID       ;PRINT TEST NUMBER AND NAME
    
```

: END OF MACRO EXPANSION OF 'PNTMAC'

8915 061116 004737 033434'  
8916 061122 103034  
8917 061124 012777 004100 117072  
8918 061132 112777 000140 117064  
8919 061140 004737 027736'  
8920 061144 103010  
8921 061146

```

JSR    PC,TINIT      ; IS A DEVICE RESET NEEDED?
BCC    30$           ; NO
MOV    #ONI!INTE,@PCSR0 ; ENABLE INTERRUPTS
MOVB   #INTE!RSET,@PCSR0 ; YES, RESET DELUA
JSR    PC,CKDNI      ; DNI ?
BCC    20$           ; YES
FTL
    
```

061146 004737 026652'

```

JSR    PC,CHKFTL     ; 'FATL' BIT SET?
    
```

8922 061152

```

ERRHRD 363.,ERR006,MSG003 ; NO, REPORT ERROR
    
```

```

061152 104456                                TRAP    C$ERHRD
061154 000553                                .WORD  363
051156 023030'                               .WORD  ERR006
061160 021736'                               .WORD  MSG003
8923 061162          ESCAPE TST                ; AND ABORT TEST
061162 104410                                TRAP    C$ESCAPE
061164 003654                                .WORD  L10053-.

8924
8925 061166 004737 030264'                   ;
8926                                     ;
8927 061172 103010                               BCC    30$
8928 061174                               FTL
                                           ; 'FATL' BIT SET?
061174 004737 026652'                               JSR    PC,CHKFTL
                                           ; YES, REPORT ERROR
8929 061200                               ERRHRD 364.,ERR006,MSG003
061200 104456                                TRAP    C$ERHRD
061202 000554                                .WORD  364
061204 023030'                               .WORD  ERR006
061206 021736'                               .WORD  MSG003
8930 061210          ESCAPE TST                ; AND ABORT TEST
061210 104410                                TRAP    C$ESCAPE
061212 003626                                .WORD  L10053 .

8931
8932 061214                                     ;
8933 061214 004737 030212'                   ;
8934 061220 004737 031732'                   ;
8935 061224 004737 032032'                   ;
8936 061230 012777 004100 116766             ;
8937 061236 112777 000101 116760             ;
8938 061244 004737 026550'                   ;
8939 061250 103010                               BCC    40$
8940 061252                               FTL
                                           ; 'FATL' BIT SET?
061252 004737 026652'                               JSR    PC,CHKFTL
                                           ; NO, REPORT ERROR
8941 061256                               ERRHRD 365.,ERR009,MSG003
061256 104456                                TRAP    C$ERHRD
061260 000555                                .WORD  365
061262 023245'                               .WORD  ERR009
061264 021736'                               .WORD  MSG003
8942 061266          ESCAPE TST                ; AND ABORT TEST
061266 104410                                TRAP    C$ESCAPE
061270 003550                                .WORD  L10053 .

8943
8944 061272 004737 030264'                   ;
8945                                     ;
8946 061276 103010                               BCC    50$
8947 061300                               FTL
                                           ; 'FATL' BIT SET?
061300 004737 026652'                               JSR    PC,CHKFTL
                                           ; YES, REPORT ERROR
8948 061304                               ERRHRD 366.,ERR006,MSG003
061304 104456                                TRAP    C$ERHRD
061306 000556                                .WORD  366
061310 023030'                               .WORD  ERR006
061312 021736'                               .WORD  MSG003
    
```



```

8949 061314          ESCAPE TST          ; AND ABORT TEST
      061314 104410
      061316 003522
                                     TRAP   C$ESCAPE
                                     .WORD  L10053-.

8950
8951          ;WRITE MODE REGISTER = INTERNAL LOOPBACK
8952
8953 061320 012705 012600' 50$:  MOV   #WTRNGS,R5          ; DEFAULT WRITE MODE FUNCTION
8954 061324 004737 032002'    JSR   PC,LDPCCB          ; LOAD FUNCTION -> PCBB
8955 061330 013737 016472' 000302'  MOV   MODE21,PCBB+2      ; MODE = INTL LOOPBACK ONLY
8956 061336 012777 004100 116660  MOV   #DNI!INTE,@PCSR0   ; ENABLE INTERRUPTS
8957 061344 112777 000102 116652  MOVB  #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
8958 061352 004737 026550'    JSR   PC,CHKDNI          ; DNI ?
8959 061356 103010          BCC   60$              ; YES
8960 061360          FTL

      061360 004737 026652'    JSR   PC,CHKFTL          ; 'FATL' BIT SET?

8961 061364          ERRHRD 367.,ERR010,MSG003 ; NO, REPORT ERROR
      061364 104456
      061366 000557
      061370 023331'
      061372 021736'
                                     TRAP   C$ERRHRD
                                     .WORD  367
                                     .WORD  ERR010
                                     .WORD  MSG003

8962 061374          ESCAPE TST          ; AND ABORT TEST
      061374 104410
      061376 003442
                                     TRAP   C$ESCAPE
                                     .WORD  L10053-.

8963
8964 061400 004737 030264' 60$:  JSR   PC,CLRDN1          ; WRITE ONE TO CLEAR DNI
8965
8966 061404 103010          BCC   70$              ; ERROR ?
8967 061406          FTL
                                     ; NO

      061406 004737 026652'    JSR   PC,CHKFTL          ; 'FATL' BIT SET?

8968 061412          ERRHRD 370.,ERR006,MSG003 ; YES, REPORT ERROR
      061412 104456
      061414 000562
      061416 023030'
      061420 021736'
                                     TRAP   C$ERRHRD
                                     .WORD  370
                                     .WORD  ERR006
                                     .WORD  MSG003

8969 061422          ESCAPE TST          ; AND ABORT TEST
      061422 104410
      061424 003414
                                     TRAP   C$ESCAPE
                                     .WORD  L10053 .

8970
8971          ;WRITE RING FORMAT
8972
8973 061426 012705 012540' 70$:  MOV   #WTRNGS,R5          ; DEFAULT WRITE RING FORMAT FUNCTION
8974 061432 004737 032002'    JSR   PC,LDPCCB          ; LOAD FUNCTION -> PCBB
8975 061436 012705 012704'    MOV   #RFRMT,R5         ; DEFAULT RING FORMAT
8976 061442 012700 000006     MOV   #6,R0             ; FORMAT = SIX WORDS
8977 061446 004737 032260'    JSR   PC,LDUDBB         ; LOAD RING FORMAT -> UDBB
8978 061452 012777 004100 116544  MOV   #DNI!INTE,@PCSR0   ; ENABLE INTERRUPTS
8979 061460 112777 000102 116536  MOVB  #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
8980 061466 004737 026550'    JSR   PC,CHKDNI          ; DNI ?
8981 061472 103010          BCC   80$              ; YES
8982 061474          FTL

      061474 004737 026652'    JSR   PC,CHKFTL          ; 'FATL' BIT SET?
    
```

```

8983 061500          ERRHRD 371.,ERR010,MSG003      ; NO, REPORT ERROR
      061500 104456
      061502 000563
      061504 023331'
      061506 021736'
      TRAP      C$ERHRD
      .WORD    371
      .WORD    ERR010
      .WORD    MSG003
8984 061510          ESCAPE TST                    ; AND ABORT TEST
      061510 104410
      061512 003326
      TRAP      C$ESCAPE
      .WORD    L10053-.
8985
8986 061514 004737 030264'      ; 80$: JSR PC,CLRDN1      ; WRITE ONE TO CLEAR DNI ERROR?
8987 061520 103010
8988 061522          FTL                          ; NO
      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
8989 061526          ERRHRD 372.,ERR006,MSG003      ; YES, REPORT ERROR
      061526 104456
      061530 000564
      061532 023030'
      061534 021736'
      TRAP      C$ERHRD
      .WORD    372
      .WORD    ERR006
      .WORD    MSG003
8990 061536          ESCAPE TST                    ; AND ABORT TEST
      061536 104410
      061540 003300
      TRAP      C$ESCAPE
      .WORD    L10053-.
8991
8992          ;WRITE PHYSICAL ADDRESS
8993
8994 061542 012705 000272'      ; 90$: MOV #DEFAULT,R5      ; GET DEFAULT PHYSICAL ADDRESS
8995 061546 004737 032050'      ; JSR PC,LDPHYA            ; SAVE IN DEFAULT TABLE
8996 061552 012705 012500'      ; MOV #WTPHYA,R5          ; DEFAULT WRITE PHYSICAL ADDR FUNC
8997 061556 004737 032002'      ; JSR PC,LDPCHB           ; LOAD FUNCTION -> PCBB
8998 061562 012777 004100 116434 ; MOV #DNI!INTE,@PCSR0    ; ENABLE INTERRUPTS
8999 061570 112777 000102 116426 ; MOVB #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
9000 061576 004737 026550'      ; JSR PC,CHKDNI          ; DNI ?
9001 061602 103010
9002 061604          FTL                          ; YES
      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
9003 061610          ERRHRD 373.,ERR010,MSG003      ; NO, REPORT ERROR
      061610 104456
      061612 000565
      061614 023331'
      061616 021736'
      TRAP      C$ERHRD
      .WORD    373
      .WORD    ERR010
      .WORD    MSG003
9004 061620          ESCAPE TST                    ; AND ABORT TEST
      061620 104410
      061622 003216
      TRAP      C$ESCAPE
      .WORD    L10053-.
9005
9006 061624 004737 030264'      ; 100$: JSR PC,CLRDN1     ; WRITE ONE TO CLEAR DNI
9007
9008 061630 103010
9009 061632          FTL                          ; ERROR ?
      BCC      102$          ; NO
      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
9010 061636          ERRHRD 374.,ERR006,MSG003      ; YES, REPORT ERROR
      061636 104456
      061640 000566
      TRAP      C$ERHRD
      .WORD    374
    
```

```

    061642 023030'
    061644 021736'
9011 051646          ESCAPE TST          ; AND ABORT TEST
    061646 104410          TRAP          C$ESCAPE
    061650 003170          .WORD          L10053-.

9012
9013          ;WRITE MULTICAST ADDRESS LIST
9014
9015 061652 012705 012520' 102$: MOV    #WTMULA,P5          ; DEFAULT WRITE MULTICAST ADDR FUNC
9016 061656 004737 032002'      JSR    PC,LDPCCB          ; LOAD FUNCTION -> PCBB
9017 061662 012705 016074'      MOV    #MULTL,R5          ; LOAD LIST INTO UDBB
9018 061666 012700 000036      MOV    #30.,R0           ; LOAD 30 ENTRIES
9019 061672 004737 032260'      JSR    PC,LDUDBB          ; MULTICAST LIST -> UDBB
9020 061676 012777 004100 116320 MOV    #DNI!INTE,@PCSR0   ; ENABLE INTERRUPTS
9021 061704 112777 000102 116312 MOVB   #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
9022 061712 004737 026550'      JSR    PC,CHKDNI          ; DNI ?
9023 061716 103010          BCC    104$              ; YES
9024 061720          FTL

    061720 004737 026652'      JSR    PC,CHKFTL          ; 'FATL' BIT SET?

9025 061724          ERRHRD 375.,ERR010,MSG003      ; NO, REPORT ERROR
    061724 104456          TRAP          C$ERHRD
    061726 000567          .WORD          375
    061730 023331'          .WORD          ERR010
    061732 021736'          .WORD          MSG003

9026 061734          ESCAPE TST          ; AND ABORT TEST
    061734 104410          TRAP          C$ESCAPE
    061736 003102          .WORD          L10053.

9027
9028 061740 004737 030264' 104$: JSR    PC,CLRDN1          ; WRITE ONE TO CLEAR DNI ERROR?
9029 061744 103010          BCC    106$              ; NO
9030 061746          FTL

    061746 004737 026652'      JSR    PC,CHKFTL          ; 'FATL' BIT SET?

9031 061752          ERRHRD 376.,ERR006,MSG003      ; YES, REPORT ERROR
    061752 104456          TRAP          C$ERHRD
    061754 000570          .WORD          376
    061756 023030'          .WORD          ERR006
    061760 021736'          .WORD          MSG003

9032 061762          ESCAPE TST          ; AND ABORT TEST
    061762 104410          TRAP          C$ESCAPE
    061764 003054          .WORD          L10053-.

9033
9034          ;DO TEN LOOPS WITH DEST ADDR = MULTICAST ADDRESS
9035
9036 061766          106$:
9037 061766 012704 000012      MOV    #10.,R4           ; DO LOOP = TEN
9038 061772 012702 016074'      MOV    #MULTL,R2          ; R2 POINTS TO MULTICAST LIST
9039 061776 012701 000272'      MOV    #DEFAULT,R1        ; SOURCE = PHYSICAL ADDRESS
9040 062002 004737 033206'      JSR    PC,SRCDST          ; STORE THIS IN TABLES

9041
9042          ;SET UP RINGS FOR ONE BUFFER LOOPBACK
9043
9044 062006 012705 014410' 110$: MOV    #TDRB1A,R5          ; DEFAULT ONE BUFFER TRANSMIT RING
9045 062012 004737 032164'      JSR    PC,LDTDRB          ; LOAD TDRB
  
```

```

9046 062016 012705 012750'      MOV    #RDRB1A,R5      ; DEFAULT ONE BUFFER RECEIVE RING
9047 062022 004737 032070'      JSR    PC,LDRDRB      ; LOAD RDRB
9048
9049          ;SET UP BUFFERS AND START
9050
9051 062026 012701 000272'      MOV    #DFault,R1     ; POINT TO SOURCE ADDRESS
9052 062032 004737 033206'      JSR    PC,SRCDST     ; R2 IS MULTICAST ADR LIST POINTER
9053 062036 005037 016562'      CLR    D0CRC         ; NO CRC APPENDED
9054 062042 012737 000006 016560'  MOV    #6,BYTCNT     ; BYTES/PACKET
9055 062050 004737 033006'      JSR    PC,SETBUF     ; SET UP BUFFERS
9056 062054 012777 004100 116142  MOV    #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
9057 062062 112777 000104 116134  MOVB   #INTE!START,@PCSR0 ; ISSUE START PORT COMMAND
9058 062070 004737 026550'      JSR    PC,CHKDNI     ; DNI?
9059 062074 103010                BCC    120#          ; YES
9060 062076                FTL

          062076 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?

9061 062102                ERRHRD 377.,ERR012,MSG003 ; NO, REPORT ERROR
          062102 104456                TRAP  C#ERHRD
          062104 000571                .WORD 377
          062106 023447'                .WORD ERR012
          062110 021736'                .WORD MSG003

9062 062112                ESCAPE TST           ; AND ABORT TEST
          062112 104410                TRAP  C#ESCAPE
          062114 002724                .WORD L10053-.

9063
9064 062116 004737 030264'      ;120#: JSR    PC,CLR DNI     ; WRITE ONE TO CLEAR DNI
9065                                ; ERROR ?
9066 062122 103010                BCC    130#          ; NO
9067 062124                FTL

          062124 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?

9068 062130                ERRHRD 380.,ERR006,MSG003 ; YES, REPORT ERROR
          062130 104456                TRAP  C#ERHRD
          062132 000574                .WORD 380
          062134 023030'                .WORD ERR006
          062136 021736'                .WORD MSG003

9069 062140                ESCAPE TST           ; AND ABORT TEST
          062140 104410                TRAP  C#ESCAPE
          062142 002676                .WORD L10053-.

9070
9071 062144 004737 027566'      ;130#: JSR    PC,CHKTXI     ; TXI ?
9072 062150 103010                BCC    140#          ; YES
9073 062152                FTL

          062152 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?

9074 062156                ERRHRD 381.,ERR013,MSG003 ; NO, REPORT ERROR
          062156 104456                TRAP  C#ERHRD
          062160 000575                .WORD 381
          062162 023530'                .WORD ERR013
          062164 021736'                .WORD MSG003

9075 062166                ESCAPE TST           ; AND ABORT TEST
          062166 104410                TRAP  C#ESCAPE
          062170 002650                .WORD L10053-.
    
```

```

9076
9077 062172 004737 030500'      ; 140$: JSR    PC,CLRTXI      ; WRITE ONE TO CLEAR TXI ERROR?
9078 062176 103010              ; BCC    150$                ; NO
9079 062200
                                ; FTL
                                ;
                                ; JSR    PC,CHKFTL          ; 'FATL' BIT SET?
                                ; ERRHRD 382.,ERR014,MSG003 ; YES, REPORT ERROR
                                ; TRAP   C$ERHRD
                                ; .WORD  382
                                ; .WORD  ERR014
                                ; .WORD  MSG003
9080 062204
                                ;
                                ; 104456
                                ; 000576
                                ; 023561'
                                ; 021736'
9081 062214
                                ; ESCAPE TST          ; AND ABORT TEST
                                ; TRAP   C$ESCAPE
                                ; .WORD  L10053-.
                                ; 104410
                                ; 002622
9082
                                ; 150$: MOV    #TDRB,R5      ; CHECK TDRB OWNERSHIP
9083 062220 012705 000620'      ; JSR    PC,CHKOWN         ; OWN = PORT DRIVER ?
9084 062224 004737 027024'      ; BCC    160$                ; YES
9085 062230 103010
                                ; FTL
                                ;
                                ; JSR    PC,CHKFTL          ; 'FATL' BIT SET?
                                ; ERRHRD 383.,ERR018      ; NO, REPORT ERROR
                                ; TRAP   C$ERHRD
                                ; .WORD  383
                                ; .WORD  ERR018
                                ; .WORD  0
9087 062236
                                ;
                                ; 104456
                                ; 000577
                                ; 024026'
                                ; 000000
9088 062246
                                ; ESCAPE TST          ; AND ABORT TEST
                                ; TRAP   C$ESCAPE
                                ; .WORD  L10053 .
                                ; 104410
                                ; 002570
9089
                                ; 160$: MOV    #TDR14A,R5    ; POINT TO EXPECTED TDRB
9090 062252 012705 016264'      ; JSR    PC,LDXTDR        ; LOAD INTO XTDRBO TABLE
9091 062256 004737 032370'      ; MOV    #TDRB,R5        ; CHECK TDRB
9092 062262 012705 000620'      ; JSR    PC,CHKTDR        ; ERRORS ?
9093 062266 004737 027500'      ; BCC    170$                ; NO
9094 062272 103010
                                ; FTL
                                ;
                                ; JSR    PC,CHKFTL          ; 'FATL' BIT SET?
                                ; ERRHRD 384.,ERR020,MSG005 ; YES, REPORT ERROR
                                ; TRAP   C$ERHRD
                                ; .WORD  384
                                ; .WORD  ERR020
                                ; .WORD  MSG005
9096 062300
                                ;
                                ; 104456
                                ; 000600
                                ; 024206'
                                ; 022042'
9097 062310
                                ; ESCAPE TST          ; AND ABORT TEST
                                ; TRAP   C$ESCAPE
                                ; .WORD  L10053-.
                                ; 104410
                                ; 002526
9098
                                ; 170$: JSR    PC,CHKRXI      ; RXI ?
9099 062314 004737 027316'      ; BCC    180$                ; YES
9100 062320 103010
                                ; FTL
                                ;
                                ; JSR    PC,CHKFTL          ; 'FATL' BIT SET?
                                ; ERRHRD 385.,ERR015,MSG003 ; NO, REPORT ERROR
9101 062322 004737 026652'
9102 062326
    
```

	062326	104456					TRAP	C#ERHRD
	062330	000601					.WORD	385
	062332	023627'					.WORD	ERR015
	062334	021736'					.WORD	MSG003
9103	062336			ESCAPE	TST			
	062336	104410					TRAP	C#ESCAPE
	062340	002500					.WORD	OF
9104								
9105	062342	004737	030432'	i	180#:	JSR	PC,CLRRXI	; WRITE ONE TO CLEAR RXI
9106								; ERROR ?
9107	062346	103010				BCC	190#	; NO
9108	062350					FTL		
	062350	004737	026652'			JSR	PC,CHKFTL	; 'FATL' BIT SET?
9109	062354					ERRHRD	386.,ERR016,MSG003	; YES, REPORT ERROR
	062354	104456					TRAP	C#ERHRD
	062356	000602					.WORD	386
	062360	023660'					.WORD	ERR016
	062362	021736'					.WORD	MSG003
9110	062364					ESCAPE	TST	; AND ABORT TEST
	062364	104410					TRAP	C#ESCAPE
	062366	002452					.WORD	L10053-.
9111								
9112	062370	012705	000660'	i	190#:	MOV	#RDRB,R5	; CHECK RDRB OWNERSHIP
9113	062374	004737	027024'			JSR	PC,CHKOWN	; OWN = PORT DRIVER ?
9114	062400	103010				BCC	200#	; YES
9115	062402					FTL		
	062402	004737	026652'			JSR	PC,CHKFTL	; 'FATL' BIT SET?
9116	062406					ERRHRD	387.,ERR017	; NO, REPORT ERROR
	062406	104456					TRAP	C#ERHRD
	062410	000603					.WORD	387
	062412	023726'					.WORD	ERR017
	062414	000000					.WORD	0
9117	062416					ESCAPE	TST	; AND ABORT TEST
	062416	104410					TRAP	C#ESCAPE
	062420	002420					.WORD	L10053-.
9118								
9119	062422	012705	016454'	i	200#:	MOV	#RDR20C,R5	; POINT TO EXPECTED RDRB
9120	062426	004737	032340'			JSR	PC,LDRDR	; LOAD INTO XRDRBO TABLE
9121	062432	012705	000660'			MOV	#RDRB,R5	; CHECK RDRB
9122	062436	004737	027206'			JSR	PC,CHKRDR	; ERRORS ?
9123	062442	103010				BCC	210#	; NO
9124	062444					FTL		
	062444	004737	026652'			JSR	PC,CHKFTL	; 'FATL' BIT SET?
9125	062450					ERRHRD	390.,ERR021,MSG006	; YES, REPORT ERROR
	062450	104456					TRAP	C#ERHRD
	062452	000606					.WORD	390
	062454	024267'					.WORD	ERR021
	062456	022204'					.WORD	MSG006
9126	062460					ESCAPE	TST	; AND ABORT TEST
	062460	104410					TRAP	C#ESCAPE
	062462	002356					.WORD	L10053-.

```

9127
9128 ;COMPARE RBUF WITH TBUF
9129
9130 062464 013705 016560' 210$: MOV BYTCNT,R5 ; COMPARE DATA
9131 062470 004737 030712' JSR PC,CMPCAT ; DATA COMPARE ERROR ?
9132 062474 103006 BCC 220$ ; NO
9133 062476 ERRHRD 391.,ERR022,MSG007 ; YES, REPORT ERROR
      062476 104456 TRAP C$ERHRD
      062500 000607 .WORD 391
      062502 024350' .WORD ERR022
      062504 022346' .WORD MSG007
9134 062506 ESCAPE TST ; AND ABORT TEST
      062506 104410 TRAP C$ESCAPE
      062510 002330 .WORD L10053-.
9135
9136 062512 ;
9137 062512 012705 006472' 220$: MOV #RBUF+26.,R5 ; CHECK CRC
9138 062516 004737 030642' JSR PC,CMPCRC ; ERRORS ?
9139 062522 103006 BCC 230$ ; NO
9140 062524 ERRHRD 392.,ERR023,MSG008 ; YES, REPORT ERROR
      062524 104456 TRAP C$ERHRD
      062526 000610 .WORD 392
      062530 024417' .WORD ERR023
      062532 022400' .WORD MSG008
9141 062534 ESCAPE TST ; AND ABORT TEST
      062534 104410 TRAP C$ESCAPE
      062536 002302 .WORD L10053-.
9142
9143 062540 ;
9144 062540 012777 004100 115456 230$: MOV #DNI!INTE,@PCSR0 ;ENABLE INTERRUPTS
9145 062546 112777 000117 115450 MOVB #INTE!STOP,@PCSR0 ; ISSUE STOP PORT COMMAND
9146 062554 004737 026550' JSR PC,CHKDNI ; DNI ?
9147 062560 103010 BCC 240$ ; YES
9148 062562 FTL
      062562 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?
9149 062566 ERRHRD 393.,ERR019,MSG003 ; NO, REPORT ERROR
      062566 104456 TRAP C$ERHRD
      062570 000611 .WORD 393
      062572 024126' .WORD ERR019
      062574 021736' .WORD MSG003
9150 062576 ESCAPE TST ; AND ABORT TEST
      062576 104410 TRAP C$ESCAPE
      062600 002240 .WORD L10053-.
9151
9152 062602 004737 030264' 240$: JSR PC,CLRDN1 ; WRITE ONE TO CLEAR DNI
9153 BCC 245$ ; ERROR ?
9154 062606 103010 FTL ; NO
9155 062610
      062610 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?
9156 062614 ERRHRD 394.,ERR006,MSG003 ; YES, REPORT ERROR
      062614 104456 TRAP C$ERHRD
      062616 000612 .WORD 394
      062620 023030' .WORD ERR006
  
```







9223	063130	012705	016344'	300\$:	MOV	@TDR21X,R5		; POINT TO EXPECTED TDRB			
9224	063134	004737	032370'		JSR	PC,LDXTDR		; LOAD INTO XTDRBC TABLE			
9225	063140	012705	000620'		MOV	@TURB,R5		; CHECK TDRB			
9226	063144	004737	027500'		JSR	PC,CHKTDR		; ERRORS ?			
9227	063150	103010			BCC	310\$		; NO			
9228	063152				FTL						
	063152	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?			
9229	063156				ERRHRD	402.,ERR020,MSG005		; YES, REPORT ERROR			
	063156	104456							TRAP	C\$ERHRD	
	063160	000622							.WORD	402	
	063162	024206'							.WORD	ERR020	
	063164	022042'							.WORD	MSG005	
9230	063166				ESCAPE	TST		; AND ABORT TEST			
	063166	104410							TRAP	C\$ESCAPE	
	063170	001650							.WORD	L10053-	
9231											
9232	063172	004737	032422'	310\$:	JSR	PC,NORXI		; RXI ?			
9233	063176	103010			BCC	320\$		; NO			
9234	063200				FTL						
	063200	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?			
9235	063204				ERRHRD	403.,ERR039		; YES, REPORT ERROR			
	063204	104456							TRAP	C\$ERHRD	
	063206	000623							WORD	403	
	063210	025575'							.WORD	ERR039	
	063212	000000							.WORD	0	
9236	063214				ESCAPE	TST		; AND ABORT TEST			
	063214	104410							TRAP	C\$ESCAPE	
	063216	001622							.WORD	L10053 .	
9237											
9238	063220			320\$:							
9239	063220	012777	004100	114776	MOV	@DNI!INTE,@PCSRO		; ENABLE INTERRUPTS			
9240	063226	112777	000117	114770	MOV	@INTE!STOP,@PCSRO		; ISSUE STOP PORT COMMAND			
9241	063234	004737	026550'		JSR	PC,CHKDNI		; DNI ?			
9242	063240	103010			BCC	330\$		; YES			
9243	063242				FTL						
	063242	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?			
9244	063246				ERRHRD	404.,ERR019,MSG003		; NO, REPORT ERROR			
	063246	104456							TRAP	C\$ERHRD	
	063250	000624							.WORD	404	
	063252	024126'							.WORD	ERR019	
	063254	021736'							.WORD	MSG003	
9245	063256				ESCAPE	TST		; AND ABORT TEST			
	063256	104410							TRAP	C\$ESCAPE	
	063260	001560							.WORD	L10053 .	
9246											
9247	063262	004737	030264'	330\$:	JSR	PC,CLRDN1		; WRITE ONE TO CLEAR DNI			
9248								; ERROR ?			
9249	063266	103010			BCC	335\$		; NO			
9250	063270				FTL						
	063270	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?			

```

9251 063274          ERRHRD  405.,ERR006,MSG003      ; YES, REPORT ERROR
      063274 104456
      063276 000625
      063300 023030'
      063302 021736'
9252 063304          ESCAPE  TST                    ; AND ABORT TEST
      063304 104410
      063306 001532
9253
9254 063310          ;
9255 063310 004737 030236'      335$: JSR    PC,CLRCV          ; CLEAR RECEIVER
9256 063314 062702 000006      ADD    #6,R2          ; UPDATE R2
9257 063320 005304          DEC    R4              ; DONE 10 LOOPBACKS?
9258 063322 001402          BEQ    340$            ; YES, EXIT LOOP
9259 063324 000137 062664'      JMP    250$            ; NO, LOOP AGAIN
9260
9261          ;
9262          ;REPEAT WITH COMPLEMENTED MULTICAST ADDRESS LIST
9263          ;
9264          ;WRITE MULTICAST ADDRESS LIST
9265 063330          340$:
9266 063330 012705 012520'      MOV    #WTMULA,R5     ; DEFAULT WRITE MULTICAST ADDR FUNC
9267 063334 004737 032002'      JSR    PC,LDPBB       ; LOAD FUNCTION -> PCBB
9268 063340 012705 016170'      MOV    #MULTLC,R5    ; LOAD LIST INTO UDBB
9269 063344 012700 000036      MOV    #30.,R0       ; LOAD 30 ENTRIES
9270 063350 004737 032260'      JSR    PC,LDUDBB     ; MULTICAST LIST -> UDBB
9271 063354 012777 004100 114642  MOV    #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
9272 063362 112777 000102 114634  MOVB   #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
9273 063370 004737 026550'      JSR    PC,CHKDNI     ; DNI ?
9274 063374 103010          BCC    350$            ; YES
9275 063376          FTL
      063376 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?
9276 063402          ERRHRD  406.,ERR010,MSG003      ; NO, REPORT ERROR
      063402 104456
      063404 000626
      063406 023331'
      063410 021736'
9277 063412          ESCAPE  TST                    ; AND ABORT TEST
      063412 104410
      063414 001424
9278
9279 063416 004737 030264'      ;
9280          350$: JSR    PC,CLRDN1          ; WRITE ONE TO CLEAR DNI
9281 063422 103010          BCC    355$            ; ERROR ?
9282 063424          FTL
      063424 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?
9283 063430          ERRHRD  407.,ERR006,MSG003      ; YES, REPORT ERROR
      063430 104456
      063432 000626
      063434 023030'
      063436 021736'
9284 063440          ESCAPE  TST                    ; AND ABORT TEST
  
```

```

063440 104410
063442 001376
9285
9286
9287
9288
9289 063444 012704 000012
9290 063450 012702 016170'
9291
9292
9293
9294 063454 012705 014410'
9295 063460 004737 032164'
9296 063464 012705 012750'
9297 063470 004737 032070'
9298
9299
9300
9301 063474 012701 000264'
9302 063500 004737 033206'
9303 063504 005037 016562'
9304 063510 012737 000006 016560'
9305 063516 004737 033006'
9306 063522 012777 004100 114474
9307 063530 112777 000104 114466
9308 063536 004737 026550'
9309 063542 103010
9310 063544
063544 004737 026652'
9311 063550
063550 104456
063552 000632
063554 023447'
063556 021736'
9312 063560
063560 104410
063562 001256
9313
9314 063564 004737 030264'
9315
9316 063570 103010
9317 063572
063572 004737 026652'
9318 063576
063576 104456
063600 000633
063602 023030'
063604 021736'
9319 063606
063606 104410
063610 001230
9320
9321 063612 004737 027566'
    
```

TRAP C\$ESCAPE  
.WORD L10053-

```

;DO TEN LOOPS WITH DEST ADDR = NEW COMPLEMENTED MULTICAST ADDRESS
;
355$: MOV #10.,R4 ; DO LOOP = TEN
      MOV #MULTLC,R2 ; R2 = COMPLEMENTED ADDRESS LIST
;
;SET UP RINGS FOR ONE BUFFER LOOPBACK
360$: MOV #TDRB1A,R5 ; DEFAULT ONE BUFFER TRANSMIT RING
      JSR PC,LDTDRB ; LOAD TDRB
      MOV #RDRB1A,R5 ; DEFAULT ONE BUFFER RECEIVE RING
      JSR PC,LDRDRB ; LOAD RDRB
;
;SET UP BUFFERS AND START
      MOV #SRC,R1 ; SOURCE = PHYSICAL ADDRESS
      JSR PC,SRCDST ; R2 = NEW COMPLEMENTED MULTICAST ADR
      CLR DDCRC ; NO APPEND CRC
      MOV #6,BYTCNT ; BYTES/PACKET
      JSR PC,SETBUF ; SET UP BUFFERS
      MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
      MOVB #INTE!START,@PCSR0 ; ISSUE START PORT COMMAND
      JSR PC,CHKDNI ; DNI?
      BCC 370$ ; YES
      FTL
      JSR PC,CHKFTL ; 'FA' BIT SET?
      ERRHRD 410.,ERR012,MSG003 ; NO, REPORT ERROR
    
```

TRAP C\$ERHRD  
.WORD 410  
.WORD ERR012  
.WORD MSG003

```

      ESCAPE TST ; AND ABORT TEST
    
```

TRAP C\$ESCAPE  
.WORD L10053-

```

;
370$: JSR PC,CLRDN1 ; WRITE ONE TO CLEAR DNI
      BCC 380$ ; ERROR ?
      FTL ; NO
      JSR PC,CHKFTL ; 'FATL' BIT SET?
      ERRHRD 411.,ERR006,MSG003 ; YES, REPORT ERROR
    
```

TRAP C\$ERHRD  
.WORD 411  
.WORD ERR006  
.WORD MSG003

```

      ESCAPE TST ; AND ABORT TEST
    
```

TRAP C\$ESCAPE  
.WORD L10053

```

;
380$: JSR PC,CHKTXI ; TXI ?
    
```

9322	063616	103010		BCC	390\$		; YES		
9323	063620			FTL					
	063620	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
9324	063624			ERRHRD	412.,ERR013,MSG003		; NO, REPORT ERROR	TRAP	C\$ERHRD
	063624	104456						.WORD	412
	063626	000634						.WORD	ERR013
	063630	023530'						.WORD	MSG003
	063632	021736'							
9325	063634			ESCAPE	TST		; AND ABORT TEST	TRAP	C\$ESCAPE
	063634	104410						.WORD	L10053--
	063636	001202							
9326									
9327	063640	004737	030500'	JSR	PC,CLRTXI	390\$:	; WRITE ONE TO CLEAR TXI		
9328							; ERROR ?		
9329	063644	103010		BCC	400\$		; NO		
9330	063646			FTL					
	063646	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
9331	063652			ERRHRD	413.,ERR014,MSG003		; YES, REPORT ERROR	TRAP	C\$ERHRD
	063652	104456						.WORD	413
	063654	000635						.WORD	ERR014
	063656	023561'						.WORD	MSG003
	063660	021736'							
9332	063662			ESCAPE	TST		; AND ABORT TEST	TRAP	C\$ESCAPE
	063662	104410						.WORD	L10053--
	063664	001154							
9333									
9334	063666	012705	000620'	MOV	#TDRB,R5	400\$:	; CHECK TDRB OWNERSHIP		
9335	063672	004737	027024'	JSR	PC,CHKOWN		; OWN = PORT DRIVER ?		
9336	063676	103010		BCC	410\$		; YES		
9337	063700			FTL					
	063700	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
9338	063704			ERRHRD	414.,ERR018		; NO, REPORT ERROR	TRAP	C\$ERHRD
	063704	104456						.WORD	414
	063706	000636						.WORD	ERR018
	063710	024026'						.WORD	0
	063712	000000							
9339	063714			ESCAPE	TST		; AND ABORT TEST	TRAP	C\$ESCAPE
	063714	104410						.WORD	L10053--
	063716	001122							
9340									
9341	063720	012705	016264'	MOV	#TDR14A,R5	410\$:	; POINT TO EXPECTED TDRB		
9342	063724	004737	032370'	JSR	PC,LDXTDR		; LOAD INTO XTDRBO TABLE		
9343	063730	012705	000620'	MOV	#TDRB,R5		; CHECK TDRB		
9344	063734	04737	027500'	JSR	PC,CHKTDR		; ERRORS ?		
9345	063740	103010		BCC	420\$		; NO		
9346	063742			FTL					
	063742	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
9347	063746			ERRHRD	415.,ERR020,MSG005		; YES, REPORT ERROR	TRAP	C\$ERHRD
	063746	104456							

	063750	000637						.WORD	415
	063752	024206'						.WORD	ERR020
	063754	022042'						.WORD	MSG005
9348	063756			ESCAPE	TST				; AND ABORT TEST
	063756	104410						TRAP	C\$ESCAPE
	063760	001060						.WORD	L10053-.
9349									
9350	063762	004737	027316'	i	420\$:	JSR	PC,CHKRXI		; RXI ?
9351	063766	103010				BCC	430\$		; YES
9352	063770					FTL			
	063770	004737	026652'			JSR	PC,CHKFTL		; 'FATL' BIT SET?
9353	063774					ERRHRD	416.,ERR015,MSG003		; NO, REPORT ERROR
	063774	104456						TRAP	C\$ERHRD
	063776	000640						.WORD	416
	064000	023627'						.WORD	ERR015
	064002	021736'						.WORD	MSG003
9354	064004					ESCAPE	TST		; AND ABORT TEST
	064004	104410						TRAP	C\$ESCAPE
	064006	001032						.WORD	L10053-.
9355									
9356	064010	004737	030432'	i	430\$:	JSR	PC,CLRRXI		; WRITE ONE TO CLEAR RXI
9357									; ERROR ?
9358	064014	103010				BCC	440\$		; NO
9359	064016					FTL			
	064016	004737	026652'			JSR	PC,CHKFTL		; 'FATL' BIT SET?
9360	064022					ERRHRD	417.,ERR016,MSG003		; YES, REPORT ERROR
	064022	104456						TRAP	C\$ERHRD
	064024	000641						.WORD	417
	064026	023660'						.WORD	ERR016
	064030	021736'						.WORD	MSG003
9361	064032					ESCAPE	TST		; AND ABORT TEST
	064032	104410						TRAP	C\$ESCAPE
	064034	001004						.WORD	L10053-.
9362									
9363	064036	012705	000660'	i	440\$:	MOV	#RDRB,R5		; CHECK RDRB OWNERSHIP
9364	064042	004737	027024'			JSR	PC,CHKOWN		; OWN = PORT DRIVER ?
9365	064046	103010				BCC	450\$		; YES
9366	064050					FTL			
	064050	004737	026652'			JSR	PC,CHKFTL		; 'FATL' BIT SET?
9367	064054					ERRHRD	420.,ERR017		; NO, REPORT ERROR
	064054	104456						TRAP	C\$ERHRD
	064056	000644						.WORD	420
	064060	023726'						.WORD	ERR017
	064062	000000						.WORD	0
9368	064064					ESCAPE	TST		; AND ABORT TEST
	064064	104410						TRAP	C\$ESCAPE
	064066	000752						.WORD	L10053-.
9369									
9370	064070	012705	016454'	i	450\$:	MOV	#RDR20C,R5		; POINT TO EXPECTED RDRB
9371	064074	004737	032340'			JSR	PC,LDXRDR		; LOAD INTO XRDRBO TABLE
9372	064100	012705	000660'			MOV	#RDRB,R5		; CHECK RDRB

```

9373 064104 004737 027206'      JSR    PC,CHKRDR      ; ERRORS ?
9374 064110 103010                BCC    460$           ; NO
9375 054112                FTL

      064112 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?
9376 064116                ERRHRD  421.,ERR021,MSG006 ; YES, REPORT ERROR
      064116 104456                TRAP   C$ERHRD
      064120 000645                .WORD  421
      064122 024267'                .WORD  ERR021
      064124 022204'                .WORD  MSG006
9377 064126                ESCAPE TST           ; AND ABORT TEST
      064126 104410                TRAP   C$ESCAPE
      064130 000710                .WORD  L10053-.

9378                ;
9379                ;COMPARE RBUF WITH TBUF
9380
9381 064132 013705 016560'      460$:  MOV    BYTCNT,R5      ; COMPARE DATA
9382 064136 004737 030712'      JSR    PC,CMPDAT     ; DATA COMPARE ERROR ?
9383 064142 103006                BCC    470$           ; NO
9384 064144                ERRHRD  422.,ERR022,MSG007 ; YES, REPORT ERROR
      064144 104456                TRAP   C$ERHRD
      064146 000646                .WORD  422
      064150 024350'                .WORD  ERR022
      064152 022346'                .WORD  MSG007
9385 064154                ESCAPE TST           ; AND ABORT TEST
      064154 104410                TRAP   C$ESCAPE
      064156 000662                .WORD  L10053-.

9386                ;
9387 064160                470$:
9388 064160 012705 006472'      MOV    $RBUF+26.,R5   ; CHECK CRC
9389 064164 004737 030642'      JSR    PC,CMPCRC     ; ERRORS ?
9390 064170 103006                BCC    480$           ; NO
9391 064172                ERRHRD  423.,ERR023,MSG008 ; YES, REPORT ERROR
      064172 104456                TRAP   C$ERHRD
      064174 000647                .WORD  423
      064176 024417'                .WORD  ERR023
      064200 022400'                .WORD  MSG008
9392 064202                ESCAPE TST           ; AND ABORT TEST
      064202 104410                TRAP   C$ESCAPE
      064204 000634                .WORD  L10053-.

9393                ;
9394 064206                480$:
9395 064206 012777 004100 114010  MOV    #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
9396 064214 112777 000117 114002  MOVB   #INTE!STOP,@PCSR0 ; ISSUE STOP PORT COMMAND
9397 064222 004737 026550'      JSR    PC,CHKDNI     ; DNI ?
9398 064226 103010                BCC    490$           ; YES
9399 064230                FTL

      064230 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?
9400 064234                ERRHRD  424.,ERR019,MSG003 ; NO, REPORT ERROR
      064234 104456                TRAP   C$ERHRD
      064236 000650                .WORD  424
      064240 024126'                .WORD  ERR019
      064242 021736'                .WORD  MSG003
9401 064244                ESCAPE TST           ; AND ABORT TEST
    
```

```

064244 104410
064246 000572
9402
9403 064250 004737 030264' ; 490$: JSR PC,CLRDNI ; WRITE ONE TO CLEAR DNI
9404 ; ERROR ?
9405 064254 103010 BCC 495$ ; NO
9406 064256 FTL

064256 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

9407 064262 ERRHRD 425.,ERR006,MSG003 ; YES, REPORT ERROR
064262 104456 TRAP C$ERHRD
064264 000651 .WORD 425
064266 023030' .WORD ERR006
064270 021736' .WORD MSG003
9408 064272 ESCAPE TST ; AND ABORT TEST
064272 104410 TRAP C$ESCAPE
064274 000544 .WORD L10053

9409 ;
9410 064276 ; 495$:
9411 064276 004737 030236' JSR PC,CLRCV ; CLEAR RECEIVE BUFFER
9412 064302 062702 000006 ADD #6,R2 ; UPDATE R2
9413 064306 062703 000004 ADD #4,R3 ; UPDATE R3
9414 064312 005304 DEC R4 ; DONE TEN LOOPBACKS ?
9415 064314 001402 BEQ 496$ ; YES
9416 064316 000137 063454' JMP 360$ ; NO
9417 ;
9418 ;DO TEN LOOPS WITH DEST ADDR = COMPLIMENT OF NEW COMPLIMENTED ADDRESS LIST
9419 ;
9420
9421 064322 012704 000012 496$: MOV #10.,R4 ; DO LOOP = TEN
9422 064326 012702 016074' MOV #MULTL,R2 ; POINT TO MULTICAST LIST
9423 ;
9424 ;SET UP RINGS FOR ONE BUFFER LOOPBACK
9425
9426 064332 012705 014410' 500$: MOV #TDRB1A,R5 ; DEFAULT ONE BUFFER TRANSMIT RING
9427 064336 004737 032164' JSR PC,LDTDRB ; LOAD TDRB
9428 064342 012705 012750' MOV #RDRB1A,R5 ; DEFAULT ONE BUFFER RECEIVE RING
9429 064346 004737 032070' JSR PC,LDRDRB ; LOAD RDRB
9430 ;
9431 ;SET UP BUFFERS AND START
9432
9433 064352 012701 000264' MOV #SRC,R1 ; SOURCE = PHYSICAL ADDRESS
9434 064356 004737 033206' JSR PC,SRCDST ; R2 = COMPLIMENT DESTINATION ADDR
9435 064362 005037 016562' CLR DOCRC ; NO APPEND CRC
9436 064366 012737 000006 016560' MOV #6,BYTCNT ; BYTES/PACKET
9437 064374 004737 033006' JSR PC,SETBUF ; SET UP BUFFERS
9438 064400 012777 004100 113616 MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
9439 064406 112777 000104 113610 MOV #INTE!START,@PCSR0 ; ISSUE START PORT COMMAND
9440 064414 004737 026550' JSR PC,CHKDNI ; DNI?
9441 064420 103010 BCC 510$ ; YES
9442 064422 FTL

064422 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

9443 064426 ERRHRD 426.,ERR012,MSG003 ; NO, REPORT ERROR
064426 104456 TRAP C$ERHRD
    
```



	064430	000652					.WORD	426
	064432	023447'					.WORD	ERR012
	064434	021736'					.WORD	MSG003
9444	064436		ESCAPE	TST				
	064436	104410					TRAP	C\$ESCAPE
	064440	000400					.WORD	L10053-
9445								
9446	064442	004737	030264'	510\$:	JSR	PC,CLRDNI		; WRITE ONE TO CLEAR DNI
9447								; ERROR ?
9448	064446	103010			BCC	520\$		; NO
9449	064450				FTL			
	064450	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
9450	064454				ERRHRD	427.,ERR006,MSG003		; YES, REPORT ERROR
	064454	104456					TRAP	C\$ERHRD
	064456	000653					.WORD	427
	064460	023030'					.WORD	ERR006
	064462	021736'					.WORD	MSG003
9451	064464		ESCAPE	TST				; AND ABORT TEST
	064464	104410					TRAP	C\$ESCAPE
	064466	000352					.WORD	L10053-
9452								
9453	064470	004737	027566'	520\$:	JSR	PC,CHKTXI		; TXI ?
9454	064474	103010			BCC	530\$		; YES
9455	064476				FTL			
	064476	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
9456	064502				ERRHRD	430.,ERR013,MSG003		; NO, REPORT ERROR
	064502	104456					TRAP	C\$ERHRD
	064504	000656					.WORD	430
	064506	023530'					.WORD	ERR013
	064510	021736'					.WORD	MSG003
9457	064512		ESCAPE	TST				; AND ABORT TEST
	064512	104410					TRAP	C\$ESCAPE
	064514	000324					.WORD	L10053 .
9458								
9459	064516	004737	030500'	530\$:	JSR	PC,CLR7XI		; WRITE ONE TO CLEAR TXI
9460								; ERROR ?
9461	064522	103010			BCC	540\$		; NO
9462	064524				FTL			
	064524	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
9463	064530				ERRHRD	431.,ERR014,MSG003		; YES, REPORT ERROR
	064530	104456					TRAP	C\$ERHRD
	064532	000657					.WORD	431
	064534	023561'					.WORD	ERR014
	064536	021736'					.WORD	MSG003
9464	064540		ESCAPE	TST				; AND ABORT TEST
	064540	104410					TRAP	C\$ESCAPE
	064542	000276					.WORD	L10053 .
9465								
9466	064544	012705	000620'	540\$:	MOV	#TDRB,R5		; CHECK TDRB OWNERSHIP
9467	064550	004737	027024'		JSR	PC,CHKOWN		; OWN = PORT DRIVER ?
9468	064554	103010			BCC	550\$		; YES

```

9459 064556          FTL
      054556 004737 026652'
9470 064562          ERRHRD 432.,ERR018          ; NO, REPORT ERROR
      064562 104456          TRAP          C$ERHRD
      064564 000660          .WORD          432
      064566 024026'          .WORD          ERR018
      064570 000000          .WORD          0
9471 064572          ESCAPE TST          ; AND ABORT TEST
      064572 104410          TRAP          C$ESCAPE
      064574 000244          .WORD          L10053-.
9472
9473 064576 012705 016344'          ; 550$: MOV #TDR21X,R5          ; POINT TO EXPECTED TDRB
9474 064602 004737 032370'          JSR PC,LDXTDR          ; LOAD INTO XTDRBO TABLE
9475 064606 012705 000620'          MOV #TDRB,R5          ; CHECK TDRB
9476 064612 004737 027500'          JSR PC,CHKTDR          ; ERRORS ?
9477 064616 103010          BCC 560$          ; NO
9478 064620          FTL
      064620 004737 026652'          JSR PC,CHKFTL          ; 'FATL' BIT SET?
9479 064624          ERRHRD 433.,ERR020,MSG005          ; YES, REPORT ERROR
      064624 104456          TRAP          C$ERHRD
      064626 000661          .WORD          433
      064630 024206'          .WORD          ERR020
      064632 022042'          .WORD          MSG005
9480 064634          ESCAPE TST          ; AND ABORT TEST
      064634 104410          TRAP          C$ESCAPE
      064636 000202          .WORD          L10053 .
9481
9482 064640 004737 032422'          ; 560$: JSR PC,NORXI          ; RXI ?
9483 064644 103010          BCC 570$          ; NO
9484 064646          FTL
      064646 004737 026652'          JSR PC,CHKFTL          ; 'FATL' BIT SET?
9485 064652          ERRHRD 434.,ERR039          ; YES, REPORT ERROR
      064652 104456          TRAP          C$ERHRD
      064654 000662          .WORD          434
      064656 025575'          .WORD          ERR039
      064660 000000          .WORD          0
9486 064662          ESCAPE TST          ; AND ABORT TEST
      064662 104410          TRAP          C$ESCAPE
      064664 000154          .WORD          L10053 .
9487
9488 064666          ; 570$:
9489 064666 012777 004100 113330          MOV #DNI!INTE,@PCSR0          ; ENABLE INTERRUPTS
9490 064674 112777 000117 113322          MOVB #INTE!STOP,@PCSR0          ; ISSUE STOP PORT COMMAND
9491 064702 004737 026550'          JSR PC,CHKDNI          ; DNI ?
9492 064706 103010          BCC 580$          ; YES
9493 064710          FTL
      064710 004737 026652'          JSR PC,CHKFTL          ; 'FATL' BIT SET?
9494 064714          ERRHRD 435.,ERR019,MSG003          ; NO, REPORT ERROR
      064714 104456          TRAP          C$ERHRD
  
```

```

064716 000663 .WORD 435
064720 024126' .WORD ERR019
054722 021736' .WORD MSG003
9495 064724 ESCAPE TST ; AND ABORT TEST
064724 104410 TRAP C#ESCAPE
064726 000112 .WORD L10053 .
9496
9497 064730 004737 030264' ;580+: JSR PC,CLRDNI ; WRITE ONE TO CLEAR DNI
9498 ; ERROR ?
9499 064734 103010 BCC 590+ ; NO
9500 064736 FTL
064736 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?
9501 064742 ERRHRD 436.,ERR006,MSG003 ; YES, REPORT ERROR
064742 104456 TRAP C#ERHRD
064744 000664 .WORD 436
064746 023030' .WORD ERR006
064750 021736' .WORD MSG003
9502 064752 ESCAPE TST ; AND ABORT TEST
064752 104410 TRAP C#ESCAPE
064754 000064 .WORD L10053-.
9503
9504 064756 ;590+: JSR PC,CLRCV ; CLEAR RECEIVE BUFFER
9505 064756 004737 030236' ADD #6,R2 ; UPDATE R2
9506 064762 062702 000006 DEC R4 ; DONE TEN LOOPBACKS ?
9507 064766 005304 BEQ 600+ ; YES, EXIT LOOP
9508 064770 001402 JMP 500+ ; NO, LOOP AGAIN
9509 064772 000137 064332' ;600+: JSR PC,CLINTR ; INSURE DELUA INTR BITS CLEAR
9510 064776
9511 064776 004737 030346'
9512
9513 065002 EXIT TST
065002 104432 TRAP C#EXIT
065004 000034 .WORD L10053-.
9514
9515 ;LOCAL TEST MESSAGE
9516
9517 065006 104 105 114 T21ID: .ASCIZ 'DELUA MULTICAST ADDRESS '
065011 125 101 040
065014 115 125 114
065017 124 111 103
065022 101 123 124
065025 040 101 104
065030 104 122 105
065033 123 123 040
065036 000
9518 .EVEN
9519
9520 065040 ENDTST
065040
065040 104401 L10053: TRAP C#ETST
  
```

9522  
9523  
9524  
9525  
9526  
9527  
9528  
9529  
9530  
9531  
9532  
9533  
9534  
9535  
9536  
9537  
9538  
9539  
9540  
9541  
9542  
9543  
9544  
9545  
9546  
9547  
9548  
9549  
9550  
9551  
9552  
9553  
9554  
9555  
9556  
9557  
9558  
9559  
9560  
9561  
9562  
9563  
9564  
9565

.SBTTL TEST 22: PROMISCUOUS ADDRESS MODE TEST

\*\*\*\*\*

THIS TEST VERIFIES THAT PROMISCUOUS ADDRESSING MODE  
IS OPERATIONAL.  
A WRITE PHYSICAL ADDRESS FUNCTION IS USED TO SET  
THE DELUA'S PHYSICAL ADDRESS.  
A WRITE MULTICAST ADDRESS LIST FUNCTION IS USED TO SET  
THE DELUA'S MULTICAST ADDRESS LIST.  
INTERNAL LOOPBACKS ARE THEN PERFORMED WITH  
CURRENTLY ENABLED AND THEN CURRENTLY DISABLED  
PHYSICAL AND MULTICAST DESTINATION ADDRESSES.

TEST SEQUENCE:

1. WRITE MODE REGISTER = INTERNAL LOOPBACK AND PROM MODE
2. WRITE RING FORMAT
3. WRITE PHYSICAL ADDRESS
4. WRITE MULTICAST ADDRESS LIST
5. SET UP RINGS AND BUFFERS  
WITH DESTINATION ADDRESS = PHYSICAL ADDRESS
6. ISSUE START
7. CHECK FOR ERRORS
8. ISSUE STOP
9. SET UP RINGS AND BUFFERS  
WITH DESTINATION ADDRESS NOT = PHYSICAL ADDRESS
10. ISSUE START
11. CHECK FOR ERRORS
12. ISSUE STOP
13. SET UP RINGS AND BUFFERS  
WITH DESTINATION ADDRESS = MULTICAST ADDRESS
14. ISSUE START
15. CHECK FOR ERRORS
16. ISSUE STOP
17. REPEAT STEPS 13 - 16 FOR ALL TEN LIST ENTRIES
18. SET UP RINGS AND BUFFERS  
WITH DESTINATION ADDRESS NOT = MULTICAST ADDRESS
19. ISSUE START
20. CHECK FOR ERRORS
21. ISSUE STOP
22. REPEAT STEPS 18 - 21 FOR ALL TEN ENTRIES

\*\*\*\*\*

9566 065042  
065042  
9567  
9568 065042

BGNTST

T22::

PNTMAC T22ID

065042 012704 071250'  
065042 004737 032734'

MOV @T22ID,R4 ;GET POINTER TO TEST NAME MESSAGE  
JSR PC,PNTID ;PRINT TEST NUMBER AND NAME

END OF MACRO EXPANSION OF 'PNTMAC'

9569 065052 004737 033434'  
9570 065056 103034  
9571 065060 012777 004100 113136

JSR PC,TINIT ; IS A DEVICE RESET NEEDED?  
BCC 30; ; NO  
MOV @DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS

9572	065066	112777	000140	113130	MOV B	0INTE!RSET,@PCSR0	; YES, RESET DELUA		
9573	065074	004737	027736'		JSR	PC,CKDNI	; DNI ?		
9574	065100	103010			BCC	20\$	; YES		
9575	065102				FTL				
	065102	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
9576	065106				ERRHRD	437.,ERR006,MSG003	; NO, REPORT ERROR		
	065106	104456						TRAP	C\$ERHRD
	065110	000665						.WORD	437
	065112	023030'						.WORD	ERR006
	065114	021736'						.WORD	MSG003
9577	065116				ESCAPE	TST	; AND ABORT TEST		
	065116	104410						TRAP	C\$ESCAPE
	065120	004160						.WORD	L10054 .
9578									
9579	065122	004737	030264'		JSR	PC,CLRDNI	; WRITE ONE TO CLEAR DNI		
9580							; ERROR ?		
9581	065126	103010			BCC	30\$	; NO		
9582	065130				FTL				
	065130	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
9583	065134				ERRHRD	440.,ERR006,MSG003	; YES, REPORT ERROR		
	065134	104456						TRAP	C\$ERHRD
	065136	000670						.WORD	440
	065140	023030'						.WORD	ERR006
	065142	021736'						.WORD	MSG003
9584	065144				ESCAPE	TST	; AND ABORT TEST		
	065144	104410						TRAP	C\$ESCAPE
	065146	004132						.WORD	L10054 .
9585									
9586	065150								
9587	065150	004737	030212'		JSR	PC,CLRBUF	; CLEAR TBUF AND RBUF		
9588	065154	004737	031732'		JSR	PC,LDDFLT	; LOAD DEFAULT PHY.ADDRESS TABLES		
9589	065160	004737	032032'		JSR	PC,LDPCSR	; ADDRESS OF PCBB -> PCSR2!3		
9590	065164	012777	004100	113032	MOV	0DNI!INTE,@PCSR0	; ENABLE INTERRUPTS		
9591	065172	112777	000101	113024	MOV B	0INTE!GETPCB,@PCSR0	; ISSUE GET_PCBB PORT COMMAND		
9592	065200	004737	026550'		JSR	PC,CKDNI	; DNI?		
9593	065204	103010			BCC	40\$	; YES		
9594	065206				FTL				
	065206	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
9595	065212				ERRHRD	441.,ERR009,MSG003	; NO, REPORT ERROR		
	065212	104456						TRAP	C\$ERHRD
	065214	000671						.WORD	441
	065216	023245'						.WORD	ERR009
	065220	021736'						.WORD	MSG003
9596	065222				ESCAPE	TST	; AND ABORT TEST		
	065222	104410						TRAP	C\$ESCAPE
	065224	004054						.WORD	L10054 -.
9597									
9598	065226	004737	030264'		JSR	PC,CLRDNI	; WRITE ONE TO CLEAR DNI		
9599							; ERROR ?		
9600	065232	103010			BCC	50\$	; NO		
9601	065234				FTL				

```

065234 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
9602 065240                ERRHRD  442.,ERR006,MSG003 ; YES, REPORT ERROR
    065240 104456                TRAP      C$ERHRD
    065242 000672                .WORD    442
    065244 023030'              .WORD    ERR006
    065246 021736'              .WORD    MSG003
9603 065250                ESCAPE  TST            ; AND ABORT TEST
    065250 104410                TRAP      C$ESCAPE
    065252 004026                .WORD    L10054-.
9604
9605      ;WRITE MODE REGISTER = INTERNAL LOOPBACK AND PROM MODE
9606
9607 065254 012705 012600'    50$:  MOV      @WTMODE,R5      ; DEFAULT WRITE MODE FUNCTION
9608 065260 004737 032002'    JSR      PC,LDPCCB        ; LOAD FUNCTION -> PCBB
9609 065264 012777 004100 112732  MOV      @DNI!INTE,@PCSR0  ; ENABLE INTERRUPTS
9610 065272 112777 000102 112724  MOVB    @INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
9611 065300 004737 026550      JSR      PC,CHKDNI        ; DNI ?
9612 065304 103010      BCC     60$              ; YES
9613 065306                FTL
065306 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
9614 065312                ERRHRD  443.,ERR010,MSG003 ; NO, REPORT ERROR
    065312 104456                TRAP      C$ERHRD
    065314 000673                .WORD    443
    065316 023331'              .WORD    ERR010
    065320 021736'              .WORD    MSG003
9615 065322                ESCAPE  TST            ; AND ABORT TEST
    065322 104410                TRAP      C$ESCAPE
    065324 003754                .WORD    L10054 .
9616
9617 065326 004737 030264'    60$:  JSR      PC,CLRDN1      ; WRITE ONE TO CLEAR DNI
9618                                ; ERROR ?
9619 065332 103010      BCC     70$              ; NO
9620 065334                FTL
065334 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
9621 065340                ERRHRD  444.,ERR006,MSG003 ; YES, REPORT ERROR
    065340 104456                TRAP      C$ERHRD
    065342 000674                .WORD    444
    065344 023030'              .WORD    ERR006
    065346 021736'              .WORD    MSG003
9622 065350                ESCAPE  TST            ; AND ABORT TEST
    065350 104410                TRAP      C$ESCAPE
    065352 003726                .WORD    L10054 .
9623
9624      ;WRITE RING FORMAT
9625
9626 065354 012705 012540'    70$:  MOV      @WTRNGS,R5      ; DEFAULT WRITE RING FORMAT FUNCTION
9627 065360 004737 032002'    JSR      PC,LDPCCB        ; LOAD FUNCTION -> PCBB
9628 065364 012705 012704'    MOV      @RFRMT,R5        ; DEFAULT RING FORMAT
9629 065370 012700 000006      MOV      @6,R0            ; FORMAT = SIX WORDS
9630 065374 004737 032260'    JSR      PC,LDUDBB        ; LOAD RING FORMAT -> UDBB
9631 065400 012777 004100 112616  MOV      @DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
  
```

```

9632 065406 112777 000102 112610      MOVB    #INTE!GETCMD,@PCSR0      ; ISSUE GET_CMD PORT COMMAND
9633 065414 004737 026550'          JSR     PC,CHKDNI                ; DNI ?
9634 065420 103010                    BCC    80$                       ; YES
9635 065422                                FTL

      065422 004737 026652'          JSR     PC,CHKFTL                ; 'FATL' BIT SET?

9636 065426                                ERRHRD  445.,ERR010,MSG003        ; NO, REPORT ERROR
      065426 104456                                TRAP   C$ERHRD
      065430 000675                                .WORD  445
      065432 023331'                                .WORD  ERR010
      065434 021736'                                .WORD  MSG003

9637 065436                                ESCAPE  TST                      ; AND ABORT TEST
      065436 104410                                TRAP   C$ESCAPE
      065440 003640                                .WORD  L10054-.

9638                                ;
9639 065442 004737 030264'          ; 80$: JSR     PC,CLRDNI          ; WRITE ONE TO CLEAR DNI
9640                                ; ERROR ?
9641 065446 103010                    BCC    90$                       ; NO
9642 065450                                FTL

      065450 004737 026652'          JSR     PC,CHKFTL                ; 'FATL' BIT SET?

9643 065454                                ERRHRD  446.,ERR006,MSG003        ; YES, REPORT ERROR
      065454 104456                                TRAP   C$ERHRD
      065456 000676                                .WORD  446
      065460 023030'                                .WORD  ERR006
      065462 021736'                                .WORD  MSG003

9644 065464                                ESCAPE  TST                      ; AND ABORT TEST
      065464 104410                                TRAP   C$ESCAPE
      065466 003612                                .WORD  L10054 .

9645                                ;
9646                                ;WRITE PHYSICAL ADDRESS
9647                                ;
9648 065470                                ; 90$:
9649 065470 012705 000272'          MOV     #DEFAULT,R5              ; GET DEFAULT PHYSICAL ADDRESS
9650 065474 004737 032050'          JSR     PC,LDPHYA                ; SAVE IN DEFAULT FILE
9651 065500 012705 012500'          MOV     #WTPHYA,R5              ; DEFAULT WRITE PHYSICAL ADDR FUNC
9652 065504 004737 032002'          JSR     PC,LDPCBB                ; LOAD FUNCTION -> PCBB
9653 065510 012777 004100 112506      MOV     #DNI!INTE,@PCSR0        ; ENABLE INTERRUPTS
9654 065516 112777 000102 112500      MOVB   #INTE!GETCMD,@PCSR0      ; ISSUE GET_CMD PORT COMMAND
9655 065524 004737 026550'          JSR     PC,CHKDNI                ; DNI ?
9656 065530 103010                    BCC    100$                      ; YES
9657 065532                                FTL

      065532 004737 026652'          JSR     PC,CHKFTL                ; 'FATL' BIT SET?

9658 065536                                ERRHRD  447.,ERR010,MSG003        ; NO, REPORT ERROR
      065536 104456                                TRAP   C$ERHRD
      065540 000677                                .WORD  447
      065542 023331'                                .WORD  ERR010
      065544 021736'                                .WORD  MSG003

9659 065546                                ESCAPE  TST                      ; AND ABORT TEST
      065546 104410                                TRAP   C$ESCAPE
      065550 003530                                .WORD  L10054-.

9660                                ;
9661 065552 004737 030264'          ; 100$: JSR     PC,CLRDNI        ; WRITE ONE TO CLEAR DNI
    
```

```

9662                                     ; ERROR ?
9663 065556 103010                      BCC 102$ ; NO
9664 065560                                     FTL

          065560 004737 026652'         JSR PC,CHKFTL ; 'FATL' BIT SET?

9665 065564                                     ERRHRD 450.,ERR006,MSG003 ; YES, REPORT ERROR
          065564 104456                                     TRAP C$ERHRD
          065566 000702                                     .WORD 450
          065570 023030'                                     .WORD ERR006
          065572 021736'                                     .WORD MSG003

9666 065574                                     ESCAPE TST ; AND ABORT TEST
          065574 104410                                     TRAP C$ESCAPE
          065576 003502                                     .WORD L10054-.

9667                                     ;
9668                                     ;WRITE MULTICAST ADDRESS LIST
9669
9670 065600 012705 012520'         102$: MOV #WTMULA,R5 ; DEFAULT WRITE MULTICAST ADDR FUNC
9671 065604 004737 032002'         JSR PC,LDPCCB ; LOAD FUNCTION -> PCBB
9672 065610 012705 016074'         MOV #MULTL,R5 ; LOAD LIST INTO UDBB
9673 065614 012700 000036'         MOV #30.,RO ; LOAD 30 ENTRIES
9674 065620 004737 032260'         JSR PC,LDUDBB ; MULTICAST LIST -> UDBB
9675 065624 012777 004100 112372   MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
9676 065632 112777 000102 112364   MOVB #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
9677 065640 004737 026550'         JSR PC,CHKDNI ; DNI ?
9678 065644 103010                      BCC 104$ ; YES
9679 065646                                     FTL

          065646 004737 026652'         JSR PC,CHKFTL ; 'FATL' BIT SET?

9680 065652                                     ERRHRD 451.,ERR010,MSG003 ; NO, REPORT ERROR
          065652 104456                                     TRAP C$ERHRD
          065654 000703                                     .WORD 451
          065656 023331'                                     .WORD ERR010
          065660 021736'                                     .WORD MSG003

9681 065662                                     ESCAPE TST ; AND ABORT TEST
          065662 104410                                     TRAP C$ESCAPE
          065664 003414                                     .WORD L10054-.

9682                                     ;
9683 065666 004737 030264'         104$: JSR PC,CLRDN1 ; WRITE ONE TO CLEAR DNI
9684                                     ; ERROR ?
9685 065672 103010                      BCC 110$ ; NO
9686 065674                                     FTL

          065674 004737 026652'         JSR PC,CHKFTL ; 'FATL' BIT SET?

9687 065700                                     ERRHRD 452.,ERR006,MSG003 ; YES, REPORT ERROR
          065700 104456                                     TRAP C$ERHRD
          065702 000704                                     .WORD 452
          065704 023030'                                     .WORD ERR006
          065706 021736'                                     .WORD MSG003

9688 065710                                     ESCAPE TST ; AND ABORT TEST
          065710 104410                                     TRAP C$ESCAPE
          065712 003366                                     .WORD L10054-.

9689                                     ;
9690                                     ;DESTINATION ADDRESS = PHYSICAL ADDRESS
9691                                     ;
    
```



```

9692          ;SET UP RINGS FOR ONE BUFFR LOOPBACK
9693
9694 065714 012705 014410' 110$: MOV    #TDRB1A,R5      ; DEFAULT ONE BUFFER TRANSMIT RING
9695 065720 004737 032164'      JSR    PC,LDTDRB      ; LOAD TDRR
9696 065724 012705 012750'      MOV    #RDRB1A,R5      ; DEFAULT ONE BUFFER RECEIVE RING
9697 065730 004737 032070'      JSR    PC,LDRDRB      ; LOAD RDRB
9698
9699          ;SET UP BUFFERS AND START
9700
9701 065734 012701 000272'      MOV    #DFault,R1      ; POINT TO SOURCE ADDRESS
9702 065740 012702 016060'      MOV    #ADR21,R2      ; DESTINATION = SOURCE
9703 065744 004737 033206'      JSR    PC,SRCDST      ; SAVE FOR PACKET BUILD
9704 065750 005037 016562'      CLR    DOCRC          ; NO APPEND CRC
9705 065754 012737 000006 016560' MOV    #6,BYTCNT      ; BYTES/PACKET
9706 065762 004737 033006'      JSR    PC,SETBUF      ; SET UP BUFFERS
9707 065766 012777 004100 112230 MOV    #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
9708 065774 112777 000104 112222 MOVB   #INTE!START,@PCSR0 ; ISSUE START PORT COMMAND
9709 066002 004737 026550'      JSR    PC,CHKONI      ; DNI?
9710 06600E 103010      BCC    120$          ; YES
9711 066010
          066010 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?
9712 066014      ERRHRD 452.,ERR012,MSG003      ; NO, REPORT ERROR
          066014 104456      TRAP    C$ERHRD
          066016 000704      .WORD 452
          066020 023447'      .WORD ERR012
          066022 021736'      .WORD MSG003
9713 066024      ESCAPE TST          ; AND ABORT TEST
          066024 104410      TRAP    C$ESCAPE
          066026 003252      .WORD L10054-.
9714
9715 066030 004737 030264' 120$: JSR    PC,CLRDNi      ; WRITE ONE TO CLEAR DNI
9716      BCC    130$          ; ERROR ?
9717 066034 103010      ; NO
9718 066036
          066036 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?
9719 066042      ERRHRD 453.,ERR006,MSG003      ; YES, REPORT ERROR
          066042 104456      TRAP    C$ERHRD
          066044 000705      .WORD 453
          066046 023030'      .WORD ERR006
          066050 021736'      .WORD MSG003
9720 066052      ESCAPE TST          ; AND ABORT TEST
          066052 104410      TRAP    C$ESCAPE
          066054 003224      .WORD L10054 .
9721
9722 066056 004737 027566' 130$: JSR    PC,CHKTXI      ; TXI ?
9723 066062 103010      BCC    140$          ; YES
9724 066064
          066064 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?
9725 066070      ERRHRD 454.,ERR013,MSG003      ; NO, REPORT ERROR
          066070 104456      TRAP    C$ERHRD
          066072 000706      .WORD 454
  
```

	066074	023530'							.WORD	ERR013
	066076	021736'							.WORD	MSG003
9726	056100			ESCAPE	TST			; AND ABORT TEST		
	066100	104410							TRAP	C\$ESCAPE
	066102	003176							.WORD	L10054-
9727										
9728	066104	004737	030500'	i	140\$:	JSR	PC,CLRTXI			; WRITE ONE TO CLEAR TXI
9729										; ERROR ?
9730	066110	103010				BCC	150\$			; NO
9731	066112					FTL				
	066112	004737	026652'			JSR	PC,CHKFTL			; 'FATL' BIT SET?
9732	066116					ERRHRD	455.,ERR014,MSG003			; YES, REPORT ERROR
	066116	104456							TRAP	C\$ERHRD
	066120	000707							.WORD	455
	066122	023561'							.WORD	ERR014
	066124	021736'							.WORD	MSG003
9733	066126					ESCAPE	TST			; AND ABORT TEST
	066126	104410							TRAP	C\$ESCAPE
	066130	003150							.WORD	L10054 .
9734										
9735	066132	012705	000620'	i	150\$:	MOV	#TDRB,R5			; CHECK TDRB OWNERSHIP
9736	066136	004737	027024'			JSR	PC,CHKOWN			; OWN = PORT DRIVER ?
9737	066142	103010				BCC	160\$			; YES
9738	066144					FTL				
	066144	004737	026652'			JSR	PC,CHKFTL			; 'FATL' BIT SET?
9739	066150					ERRHRD	456.,ERR018			; NO, REPORT ERROR
	066150	104456							TRAP	C\$ERHRD
	066152	000710							.WORD	456
	066154	024026'							.WORD	ERR018
	066156	000000							.WORD	0
9740	066160					ESCAPE	TST			; AND ABORT TEST
	066160	104410							TRAP	C\$ESCAPE
	066162	003116							.WORD	L10054 .
9741										
9742	066164	012705	016264'	i	160\$:	MOV	#TDR14A,R5			; POINT TO EXPECTED TDRB
9743	066170	004737	032370'			JSR	PC,LDXTDR			; LOAD INTO XTDRBO TABLF
9744	066174	012705	000620'			MOV	#TDRB,R5			; CHECK TDRB
9745	066200	004737	027500'			JSR	PC,CHKTDR			; ERRORS ?
9746	066204	103010				BCC	170\$			; NO
9747	066206					FTL				
	066206	004737	026652'			JSR	PC,CHKFTL			; 'FATL' BIT SET?
9748	066212					ERRHRD	457.,ERR020,MSG005			; YES, REPORT ERROR
	066212	104456							TRAP	C\$ERHRD
	066214	000711							.WORD	457
	066216	024206'							.WORD	ERR020
	066220	022042'							.WORD	MSG005
9749	066222					ESCAPE	TST			; AND ABORT TEST
	066222	104410							TRAP	C\$ESCAPE
	066224	003054							.WORD	L10054--
9750										
9751	066226	004737	027316'	i	170\$:	JSR	PC,CHKRXI			; RXI ?

9752	066232	103010		BCC	180\$		; YES		
9753	066234			FTL					
	066234	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
9754	066240			ERRHRD	460.,ERR015,MSG003		; NO, REPORT ERROR	TRAP	C\$ERHRD
	066240	104456						.WORD	460
	066242	000714						.WORD	ERR015
	066244	023627'						.WORD	MSG003
	066246	021736'							
9755	066250			ESCAPE	TST		; AND ABORT TEST	TRAP	C\$ESCAPE
	066250	104410						.WORD	L10054 .
	066252	003026							
9756									
9757	066254	004737	030432'	JSR	PC,CLRRXI	180\$:	; WRITE ONE TO CLEAR RXI		
9758							; ERROR ?		
9759	066260	103010		BCC	190\$		; NO		
9760	066262			FTL					
	066262	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
9761	066266			ERRHRD	461.,ERR016,MSG003		; YES, REPORT ERROR	TRAP	C\$ERHRD
	066266	104456						.WORD	461
	066270	000715						.WORD	ERR016
	066272	023660'						.WORD	MSG003
	066274	021736'							
9762	066276			ESCAPE	TST		; AND ABORT TEST	TRAP	C\$ESCAPE
	066276	104410						.WORD	L10054 .
	066300	003000							
9763									
9764	066302	012705	000660'	MOV	#RDRB,R5	190\$:	; CHECK RDRB OWNERSHIP		
9765	066306	004737	027024'	JSR	PC,CHKOWN		; OWN = PORT DRIVER ?		
9766	066312	103010		BCC	200\$		; YES		
9767	066314			FTL					
	066314	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
9768	066320			ERRHRD	462.,ERR017		; NO, REPORT ERROR	TRAP	C\$ERHRD
	066320	104456						.WORD	462
	066322	000716						.WORD	ERR017
	066324	023726'						.WORD	0
	066326	000000							
9769	066330			ESCAPE	TST		; AND ABORT TEST	TRAP	C\$ESCAPE
	066330	104410						.WORD	L10054 .
	066332	002746							
9770									
9771	066334	012705	016454'	MOV	#RDR20C,R5	200\$:	; POINT TO EXPECTED RDRB		
9772	066340	004737	032340'	JSR	PC,LDXRDR		; LOAD INTO XRDRBO TABLE		
9773	066344	012705	000660'	MOV	#RDRB,R5		; CHECK RDRB		
9774	066350	004737	027206'	JSR	PC,CHKRD?		; ERRORS ?		
9775	066354	103010		BCC	210\$		; NO		
9776	066356			FTL					
	066356	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
9777	066362			ERRHRD	463.,ERR021,MSG006		; YES, REPORT ERROR	TRAP	C\$ERHRD
	066362	104456							

```

066364 000717 .WORD 463
066366 024267' .WORD ERR021
056370 022204' .WORD MSG006
9778 066372 ESCAPE TST ; AND ABORT TEST TRAP C$ESCAPE
066372 104410 .WORD L10054-.
066374 002704

9779
9780 ;COMPARE RBUF WITH TBUF
9781
9782 066376 013705 016560' 210$: MOV BYTCNT,R5 ; COMPARE DATA
9783 066402 004737 030712' JSR PC,CMPDAT ; DATA COMPARE ERROR ?
9784 066406 103006 BCC 220$ ; NO
9785 066410 ERRHRD 464.,ERR022,MSG007 ; YES, REPORT ERROR TRAP C$ERHRD
066410 104456 .WORD 464
066412 000720 .WORD ERR022
066414 024350' .WORD MSG007
066416 022346'

9786 066420 ESCAPE TST ; AND ABORT TEST TRAP C$ESCAPE
066420 104410 .WORD L10054-.
066422 002656

9787
9788 ;
9789 066424 012705 006472' 220$: MOV #RBUF+26.,R5 ; CHECK CRC
9790 066430 004737 030642' JSR PC,CMPCRC ; ERRORS ?
9791 066434 103006 BCC 230$ ; NO
9792 066436 ERRHRD 465.,ERR023,MSG008 ; YES, REPORT ERROR TRAP C$ERHRD
066436 104456 .WORD 465
066440 000721 .WORD ERR023
066442 024417' .WORD MSG008
066444 022400'

9793 066446 ESCAPE TST ; AND ABORT TEST TRAP C$ESCAPE
066446 104410 .WORD L10054-.
066450 002630

9794
9795 ;
9796 066452 012777 004100 111544 230$: MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
9797 066460 112777 000117 111536 MOVB #INTE!STOP,@PCSR0 ; ISSUE STOP PORT COMMAND
9798 066466 004737 026550' JSR PC,CHKDNI ; DNI ?
9799 066472 103010 BCC 240$ ; YES
9800 066474 FTL

066474 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

9801 066500 ERRHRD 466.,ERR019,MSG003 ; NO, REPORT ERROR TRAP C$ERHRD
066500 104456 .WORD 466
066502 000722 .WORD ERR019
066504 024126' .WORD MSG003
066506 021736'

9802 066510 ESCAPE TST ; AND ABORT TEST TRAP C$ESCAPE
066510 104410 .WORD L10054 .
066512 002566

9803
9804 066514 004737 030264' 240$: JSR PC,CLRDN1 ; WRITE ONE TO CLEAR DNI
9805 ; ERROR ?
9806 066520 103010 BCC 250$ ; NO
9807 066522 FTL
    
```

```

066522 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?
9808 066526      ERRHRD 467.,ERR006,MSG003 ; YES, REPORT ERROR
      066526 104456      TRAP   C$ERHRD
      066530 000723      .WORD 467
      066532 023030      .WORD ERR006
      066534 021736'      .WORD MSG003
9809 066536      ESCAPE TST          ; AND ABORT TEST
      066536 104410      TRAP   C$ESCAPE
      066540 002540      .WORD L10054-.
9810      ;
9811      ;DESTINATION ADDRESS NOT = PHYSICAL ADDRESS
9812      ;
9813      ;SET UP RINGS FOR ONE BUFFER LOOPBACK
9814
9815 066542      250$:
9816 066542 004737 030236'      JSR    PC,CLRCV      ; CLEAR RECEIVE BUFFER
9817 066546 012705 014410'      MOV    #TDRB1A,R5   ; DEFAULT ONE BUFFER TRANSMIT RING
9818 066552 004737 ^32164'      JSR    PC,LDTDRB    ; LOAD TDRB
9819 066556 012705 012750'      MOV    #RDRB1A,R5   ; DEFAULT ONE BUFFER RECEIVE RING
9820 066562 004737 032070'      JSR    PC,LDRDRB    ; LOAD RDRB
9821
9822      ;SET UP BUFFERS AND START
9823
9824 066566 012701 000272'      MOV    #DFault,R1   ; SOURCE = PHYSICAL ADDRESS
9825 066572 012702 016066'      MOV    #ADR21C,R2   ; DEST = COMPLEMENTED ADDRESS
9826 066576 004737 033206'      JSR    %C,SRCDST    ; SAVE FOR PACKET BUILD
9827 066602 005037 016562'      CLR    D0CRC        ; NO APPEND CRC
9828 066606 012737 000006 016560'  MOV    #6,BYTCNT    ; BYTES/PACKET
9829 066614 004737 033006'      JSR    PC,SETBUF    ; SET UP BUFFERS
9830 066620 012777 004100 111376  MOV    #DNI!INTE,%PCSR0 ; ENABLE INTERRUPTS
9831 066626 112777 000104 111370  MOVB   #INTE!START,%PCSR0 ; ISSUE START PORT COMMAND
9832 066634 004737 026550'      JSR    PC,CHKDNI    ; DNI?
9833 066640 103010      BCC    260$         ; YES
9834 066642      FTL
      066642 004737 026652'      JSR    PC,CHKFTL    ; 'FATL' BIT SET?
9835 066646      ERRHRD 470.,ERR012,MSG003 ; NO, REPORT ERROR
      066646 104456      TRAP   C$ERHRD
      066650 000726      .WORD 470
      066652 023447'      .WORD ERR012
      066654 021736'      .WORD MSG003
9836 066656      ESCAPE TST          ; AND ABORT TEST
      066656 104410      TRAP   C$ESCAPE
      066660 002420      .WORD L10054 .
9837
9838 066662 004737 030264'      260$: JSR    PC,CLRDN1    ; WRITE ONE TO CLEAR DNI
9839      ; ERROR ?
9840 066666 103010      BCC    270$         ; NO
9841 066670      FTL
      066670 004737 026652'      JSR    PC,CHKFTL    ; 'FATL' BIT SET?
9842 066674      ERRHRD 471.,ERR006,MSG003 ; YES, REPORT ERROR
      066674 104456      TRAP   C$ERHRD
      066676 000727      .WORD 471
  
```

	066700	023030'					.WORD	ERR006
	066702	021736'					.WORD	MSG003
9843	056704		ESCAPE	TST		; AND ABORT TEST		
	066704	104410					TRAP	C\$ESCAPE
	066706	002372					.WORD	L10054-
9844								
9845	066710	004737	270\$:	JSR	PC,CHKTXI	; TXI ?		
9846	066714	103010		BCC	280\$	; YES		
9847	066716			FTL				
	066716	004737		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
9848	066722			ERRHRD	472.,ERR013,MSG003	; NO, REPORT ERROR		
	066722	104456					TRAP	C\$ERHRD
	066724	000730					.WORD	472
	066726	023530'					.WORD	ERR013
	066730	021736'					.WORD	MSG003
9849	066732		ESCAPE	TST		; AND ABORT TEST		
	066732	104410					TRAP	C\$ESCAPE
	066734	002344					.WORD	L10054 .
9850								
9851	066736	004737	280\$:	JSR	PC,CLRTXI	; WRITE ONE TO CLEAR TXI		
9852						; ERROR ?		
9853	066742	103010		BCC	290\$	; NO		
9854	066744			FTL				
	066744	004737		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
9855	066750			ERRHRD	473.,ERR014,MSG003	; YES, REPORT ERROR		
	066750	104456					TRAP	C\$ERHRD
	066752	000731					.WORD	473
	066754	023561'					.WORD	ERR014
	066756	021736'					.WORD	MSG003
9856	066760		ESCAPE	TST		; AND ABORT TEST		
	066760	104410					TRAP	C\$ESCAPE
	066762	002316					.WORD	L10054 .
9857								
9858	066764	012705	290\$:	MOV	@TDRB,R5	; CHECK TDRB OWNERSHIP		
9859	066770	004737		JSR	PC,CHKOWN	; OWN = PORT DRIVER ?		
9860	066774	103010		BCC	300\$	; YES		
9861	066776			FTL				
	066776	004737		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
9862	067002			ERRHRD	474.,ERR018	; NO, REPORT ERROR		
	067002	104456					TRAP	C\$ERHRD
	067004	000732					.WORD	474
	067006	024026'					.WORD	ERR018
	067010	000000					.WORD	0
9863	067012		ESCAPE	TST		; AND ABORT TEST		
	067012	104410					TRAP	C\$ESCAPE
	067014	002264					.WORD	L10054 .
9864								
9865	067016	012705	300\$:	MOV	@TDR14A,R5	; POINT TO EXPECTED TDRB		
9866	067022	004737		JSR	PC,LDXTDR	; LOAD INTO XTDRBO TABLE		
9867	067026	012705		MOV	@TDRB,R5	; CHECK TDRB		
9868	067032	004737		JSR	PC,CHKTDR	; ERRORS ?		

9869	067036	103010		BCC	310\$		; NO		
9870	067040			FTL					
	067040	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
9971	057044			ERRHRD	475.,ERR020,MSG005		; YES, REPORT ERROR		
	067044	104456						TRAP	C\$ERHRD
	067046	000733						.WORD	475
	067050	024206'						.WORD	ERR020
	067052	022042'						.WORD	MSG005
9872	067054			ESCAPE	TST		; AND ABORT TEST		
	067054	104410						TRAP	C\$ESCAPE
	067056	002222						.WORD	L10054-.
9873									
9874	067060	004737	027316'	JSR	PC,CHKRXI	310\$:	; RXI ?		
9875	067064	103010		BCC	320\$		; YES		
9876	067066			FTL					
	067066	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
9877	067072			ERRHRD	476.,ERR015,MSG003		; NO, REPORT ERROR		
	067072	104456						TRAP	C\$ERHRD
	067074	000734						.WORD	476
	067076	023627'						.WORD	ERR015
	067100	021736'						.WORD	MSG003
9878	067102			ESCAPE	TST		; AND ABORT TEST		
	067102	104410						TRAP	C\$ESCAPE
	067104	002174						.WORD	L10054-.
9879									
9880	067106	004737	030432'	JSR	PC,CLPRXI	320\$:	; WRITE ONE TO CLEAR RXI		
9881							; ERROR ?		
9882	067112	103010		BCC	330\$		; NO		
9883	067114			FTL					
	067114	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
9884	067120			ERRHRD	477.,ERR016,MSG003		; YES, REPORT ERROR		
	067120	104456						TRAP	C\$ERHRD
	067122	000735						.WORD	477
	067124	023660'						.WORD	ERR016
	067126	021736'						.WORD	MSG003
9885	067130			ESCAPE	TST		; AND ABORT TEST		
	067130	104410						TRAP	C\$ESCAPE
	067132	002146						.WORD	L10054-.
9886									
9887	067134	012705	000660'	MO	478\$,PE	330\$:	; CHECK RDRB OWNERSHIP		
9888	067140	004737	027024'	JS	P,CHK-JWA		; OWN = PORT DRIVER ?		
9889	067144	103010		BCL	340\$		; YES		
9890	067146			FTL					
	067146	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
9891	067152			ERRHRD	480.,ERR017		; NO, REPORT ERROR		
	067152	104456						TRAP	C\$ERHRD
	067154	000740						.WORD	480
	067156	023726'						.WORD	ERR017
	067160	000000						.WORD	0

```

9892 067162          ESCAPE TST          ; AND ABORT TEST
      067162 104410          TRAP          C:ESCAPE
      057164 002114          .WORD          L10054-.

9893          ;
9894 067166 012705 016454' 340: MOV    #RDR20C,R5          ; POINT TO EXPECTED RDRB
9895 067172 004737 032340' JSR    PC,LDXRDR          ; LOAD INTO XRDRBO TABLE
9896 067176 012705 000660' MOV    #RDRB,R5          ; CHECK RDRB
9897 067202 004737 027206' JSR    PC,CHKRDR          ; ERRORS ?
9898 067206 103010          BCC    350:              ; NO
9899 067210          FTL

      067210 004737 026652' JSR    PC,CHKFTL          ; 'FATL' BIT SET?

9900 067214          ERRHRD 481.,ERR021,MSG006 ; YES, REPORT ERROR
      067214 104456          TRAP          C:ERHRD
      067216 000741          .WORD          481
      067220 024267'          .WORD          ERR021
      067222 022204'          .WORD          MSG006

9901 067224          ESCAPE TST          ; AND ABORT TEST
      067224 104410          TRAP          C:ESCAPE
      067226 002052          .WORD          L10054 .

9902          ;
9903          ;COMPARE RBUF WITH TBUF
9904          ;
9905 067230 013705 016560' 350: MOV    BYTCNT,R5          ; COMPARE DATA
9906 067234 004737 030712' JSR    PC,CMPDAT          ; DATA COMPARE ERROR ?
9907 067240 103006          BCC    370:              ; NO
9908 067242          ERRHRD 482.,ERR022,MSG007 ; YES, REPORT ERROR
      067242 104456          TRAP          C:ERHRD
      067244 000742          .WORD          482
      067246 024350'          .WORD          ERR022
      067250 022346'          .WORD          MSG007

9909 067252          ESCAPE TST          ; AND ABORT TEST
      067252 104410          TRAP          C:ESCAPE
      067254 002024          .WORD          L10054 .

9910          ;
9911 067256          ;
9912 067256 012705 006472' 360: MOV    #RBUF+26.,R5          ; CHECK CRC
9913 067262 004737 030642' JSR    PC,CMPCRC          ; ERRORS ?
9914 067266 103006          BCC    370:              ; NO
9915 067270          ERRHRD 483.,ERR023,MSG008 ; YES, REPORT ERROR
      067270 104456          TRAP          C:ERHRD
      067272 000743          .WORD          483
      067274 024417'          .WORD          ERR023
      067276 022400'          .WORD          MSG008

9916 067300          ESCAPE TST          ; AND ABORT TEST
      067300 104410          TRAP          C:ESCAPE
      067302 001776          .WORD          L10054 .

9917          ;
9918 067304          ;
9919 067304 012777 004100 110712 370: MOV    #DNI!INTE,#PCSR0          ; ENABLE INTERRUPTS
9920 067312 112777 000117 110704 MOVB   #INTE!STOP,#PCSR0 ; ISSUE STOP PORT COMMAND
9921 067320 004737 026550' JSR    PC,CHKDNI          ; DNI ?
9922 067324 103010          BCC    380:              ; YES
9923 067326          FTL

      067326 004737 026652' JSR    PC,CHKFTL          ; 'FATL' BIT SET?
    
```



```

9924 067332          ERRHRD  484.,ERR019,MSG003      ; NO. REPORT ERROR
      067332 104456
      067334 000744
      067336 024126'
      067340 021736'
9925 067342          ESCAPE  TST                  ; AND ABORT TEST
      067342 104410
      067344 001734
9926
9927 067346 004737 030264' 380: JSR    PC,CLRDNI          ; WRITE ONE TO CLEAR DNI
9928
9929 067352 103010          BCC    390:          ; ERROR ?
9930 067354          FTL
      067354 004737 026652' JSR    PC,CHKFTL          ; 'FATL' BIT SET?
9931 067360          ERRHRD  485.,ERR006,MSG003      ; YES. REPORT ERROR
      067360 104456
      067362 000745
      067364 023030'
      067366 021736'
9932 067370          ESCAPE  TST                  ; AND ABORT TEST
      067370 104410
      067372 001706
9933
9934
9935
9936 067374          ;
9937 067374 012705 000272' 390: MOV    #DEFAULT,R5          ; GET DEFAULT PHYSICAL ADDRESS
9938 067400 004737 032050' JSR    PC,LDPHYA          ; SAVE IT IN DEFAULT TABLE
9939 067404 012705 012500' MOV    #WTPHYA,R5        ; DEFAULT WRITE PHYSICAL ADDR FUNC
9940 067410 004737 032002' JSR    PC,LPCBB          ; LOAD FUNCTION -> PCBB
9941 067414 012777 004100 110602 MOV    #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
9942 067422 112777 000102 110574 MOVB   #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
9943 067430 004737 026550' JSR    PC,CHKDNI          ; DNI ?
9944 067434 103010          BCC    400:          ; YES
9945 067436          FTL
      067436 004737 026652' JSR    PC,CHKFTL          ; 'FATL' BIT SET?
9946 067442          ERRHRD  486.,ERR010,MSG003      ; NO. REPORT ERROR
      067442 104456
      067444 000746
      067446 023331'
      067450 021736'
9947 067452          ESCAPE  TST                  ; AND ABORT TEST
      067452 104410
      067454 001624
9948
9949 067456 004737 030264' 400: JSR    PC,CLRDNI          ; WRITE ONE TO CLEAR DNI
9950
9951 067462 103010          BCC    410:          ; ERROR ?
9952 067464          FTL
      067464 004737 026652' JSR    PC,CHKFTL          ; 'FATL' BIT SET?
    
```

```

9953 067470          ERRHRD 487.,ERR006,MSG003      ; YES, REPORT ERROR
      067470 104456
      067472 000747
      067474 023030'
      067476 021736'
      067500          ESCAPE TST                  ; AND ABORT TEST
      067500 104410
      067502 001576
9955
9956      ;DO TEN LOOPS WITH DEST ADDR = MULTICAST ADDRESS
9957
9958 067504          410$:
9959 067504 004737 030236'      JSR      PC,CLRCV          ; CLEAR RECEIVE BUFFER
9960 067510 012704 000012      MOV      #10.,R4          ; DO LOOP = TEN
9961 067514 012702 016074'      MOV      @MULTL,R2       ; R2 POINTS TO MULTICAST LIST
9962
9963      ;SET UP RINGS FOR ONE BUFFER LOOPBACK
9964
9965 067520          420$:
9966 067520 012705 014410'      MOV      @TDRB1A,R5      ; DEFAULT ONE BUFFER TRANSMIT RING
9967 067524 004737 032164'      JSR      PC,LDTDRB      ; LOAD TDRB
9968 067530 012705 012750'      MOV      @RDRB1A,R5     ; DEFAULT ONE BUFFER RECEIVE RING
9969 067534 004737 032070'      JSR      PC,LDRDRB      ; LOAD RDRB
9970
9971      ;SET UP BUFFERS AND START
9972
9973 067540 012701 000264'      MOV      @SRC,R1         ; SOURCE = DEFAULT PHYS. ADDRESS
9974 067544 004737 033206'      JSR      PC,SRCDST      ; DEST = MULTICAST ADDRESS
9975 067550 005037 016562'      CLR      DDCRC         ; NO APPEND CRC
9976 067554 012737 000006 016560'  MOV      #6,BYTCNT      ; BYTES/PACKET
9977 067562 004737 033006'      JSR      PC,SETPUF      ; SET UP BUFFERS
9978 067566 012777 004100 110430  MOV      @DNI!INTE,@PCSRO ; ENABLE INTERRUPTS
9979 067574 112777 000104 110422  MOVB     @INTE!START,@PCSRO ; ISSUE START PORT COMMAND
9980 067602 004737 026550'      JSR      PC,CHKDNI      ; DNI?
9981 067606 103010          BCC      430$           ; YES
9982 067610          FTL
      067610 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
9983 067614          ERRHRD 490.,ERR012,MSG003      ; NO, REPORT ERROR
      067614 104456
      067616 000752
      067620 023447'
      067622 021736'
      067624          ESCAPE TST                  ; AND ABORT TEST
      067624 104410
      067626 001452
9985
9986 067630 004737 030264'      430$: JSR      PC,CLRDN1      ; WRITE ONE TO CLEAR DNI
9987
9988 067634 103010          BCC      440$           ; ERROR ?
9989 067636          FTL
      067636 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?
9990 067642          ERRHRD 491.,ERR006,MSG003      ; YES, REPORT ERROR
      067642 104456
      TRAP      C$ERHRD
    
```

	067644	000753					.WORD	491
	067646	023030'					.WORD	ERR006
	057650	021736'					.WORD	MSG003
9991	067652		ESCAPE	TST				
	067652	104410					TRAP	C\$ESCAPE
	067654	001424					.WORD	L10054 .
9992								
9993	067656	004737	440\$:	JSR	PC,CHKTXI			; TXI ?
9994	067662	103010		BCC	450\$			; YES
9995	067664			FTL				
	067664	004737		JSR	PC,CHKFTL			; 'FATL' BIT SET?
9996	067670			ERRHRD	492.,ERR013,MSG003			; NO, REPORT ERROR
	067670	104456					TRAP	C\$ERHRD
	067672	000754					.WORD	492
	067674	023530'					.WORD	ERR013
	067676	021736'					.WORD	MSG003
9997	067700			ESCAPE	TST			; AND ABORT TEST
	067700	104410					TRAP	C\$ESCAPE
	067702	001376					.WORD	L10054-.
9998								
9999	067704	004737	450\$:	JSR	PC,CLRTXI			; WRITE ONE TO CLEAR TXI
10000								; ERROR ?
10001	067710	103010		BCC	470\$			; NO
10002	067712			FTL				
	067712	004737		JSR	PC,CHKFTL			; 'FATL' BIT SET?
10003	067716			ERRHRD	493.,ERR014,MSG003			; YES, REPORT ERROR
	067716	104456					TRAP	C\$ERHRD
	067720	000755					.WORD	493
	067722	023561'					.WORD	ERR014
	067724	021736'					.WORD	MSG003
10004	067726			ESCAPE	TST			; AND ABORT TEST
	067726	104410					TRAP	C\$ESCAPE
	067730	001350					.WORD	L10054-.
10005								
10006	067732	012705	470\$:	MOV	#TDRB,R5			; CHECK TDRB OWNERSHIP
10007	067736	004737		JSR	PC,CHKOWN			; OWN = PORT DRIVER ?
10008	067742	103010		BCC	480\$			; YES
10009	067744			FTL				
	067744	004737		JSR	PC,CHKFTL			; 'FATL' BIT SET?
10010	067750			ERRHRD	494.,ERR018			; NO, REPORT ERROR
	067750	104456					TRAP	C\$ERHRD
	067752	000756					.WORD	494
	067754	024026'					.WORD	ERR018
	067756	000000					.WORD	0
10011	067760			ESCAPE	TST			; AND ABORT TEST
	067760	104410					TRAP	C\$ESCAPE
	067762	001316					.WORD	L10054 .
10012								
10013	067764	012705	480\$:	MOV	#TDR14A,R5			; POINT TO EXPECTED TDRB
10014	067770	004737		JSR	PC,LDXTDR			; LOAD INTO XTDRBO TABLE
10015	067774	012705		MOV	#TDRB,R5			; CHECK TDRB

10016	070000	004737	027500'		JSR	PC,CHKTDR		; ERRORS ?		
10017	070004	103010			BCC	490\$		; NO		
10018	070006				FTL					
	070006	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
10019	070012				ERRHRD	495.,ERR020,MSG005		; YES, REPORT ERROR		
	070012	104456							TRAP	C\$ERHRD
	070014	000757							.WORD	495
	070016	024206'							.WORD	ERR020
	070020	022042'							.WORD	MSG005
10020	070022				ESCAPE	TST		; AND ABORT TEST		
	070022	104410							TRAP	C\$ESCAPE
	070024	001254							.WORD	L10054 .
10021										
10022	070026	004737	027316'	i	JSR	PC,CHKRXI		; RXI ?		
10023	070032	103010		490\$:	BCC	500\$		; YES		
10024	070034				FTL					
	070034	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
10025	070040				ERRHRD	496.,ERR015,MSG003		; NO, REPORT FRROR		
	070040	104456							TRAP	C\$ERHRD
	070042	000760							.WORD	496
	070044	023627'							.WORD	ERR015
	070046	021736'							.WORD	MSG003
10026	070050				ESCAPE	TST		; AND ABORT TEST		
	070050	104410							TRAP	C\$ESCAPE
	070052	001226							.WORD	L10054--
10027										
10028	070054	004737	030432'	i	JSR	PC,CLRXXI		; WRITE ONE TO CLEAR RXI		
10029				500\$:				; ERROR ?		
10030	070060	103010			BCC	510\$		; NO		
10031	070062				FTL					
	070062	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
10032	070066				ERRHRD	497.,ERR016,MSG003		; YES, REPORT ERROR		
	070066	104456							TRAP	C\$ERHRD
	070070	000761							.WORD	497
	070072	023660'							.WORD	ERR016
	070074	021736'							.WORD	MSG003
10033	070076				ESCAPE	TST		; AND ABORT TEST		
	070076	104410							TRAP	C\$ESCAPE
	070100	001200							.WORD	L10054 .
10034										
10035	070102	012705	000660'	i	MOV	RDRB,R5		; CHECK RDRB OWNERSHIP		
10036	070106	004737	027024'	510\$:	JSR	PC,CHKOWN		; OWN = PORT DRIVER ?		
10037	070112	103010			BCC	520\$		; YES		
10038	070114				FTL					
	070114	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?		
10039	070120				ERRHRD	500.,ERR017		; NO, REPORT ERROR		
	070120	104456							TRAP	C\$EKHRD
	070122	000764							.WORD	500
	070124	023726'							.WORD	ERR017

```

10040 070126 000000          ESCAPE TST          ; AND ABORT TEST          .WORD 0
      070130          ;
      070130 104410          ;
      070132 001146          ;
10041 070134 012705 016454' 520$: MOV    @RDR20C,R5      ; POINT TO EXPECTED RDRB
10042 070140 004737 032340'   JSR    PC,LDXRDR      ; LOAD INTO XRDRB0 TABLE
10043 070144 012705 000660'   MOV    @RDRB,R5      ; CHECK RDRB
10044 070150 004737 027206'   JSR    PC,CHKRDR     ; ERRORS ?
10045 070154 103010          BCC    530$          ; NO
10046 070156          FTL
      070156 004737 026652'   JSR    PC,CHKFTL     ; 'FATL' BIT SET?
10048 070162          ERRHRD 501.,ERR021,MSG006 ; YES, REPORT ERROR
      070162 104456          ;
      070164 000765          ;
      070166 024267'         ;
      070170 022204'         ;
10049 070172          ESCAPE TST          ; AND ABORT TEST          TRAP C$ESCAPE
      070172 104410          ;
      070174 001104          ;
10050          ;
10051          ;COMPARE RBUF WITH TBUF
10052          ;
10053 070176 013705 016560' 530$: MOV    BYTCNT,R5      ; COMPARE DATA
10054 070202 004737 030712'   JSR    PC,CMPDAT     ; DATA COMPARE ERROR ?
10055 070206 103006          BCC    540$          ; NO
10056 070210          ERRHRD 502.,ERR022,MSG007 ; YES, REPORT ERROR
      070210 104456          ;
      070212 000766          ;
      070214 024350'         ;
      070216 022346'         ;
10057 070220          ESCAPE TST          ; AND ABORT TEST          TRAP C$ESCAPE
      070220 104410          ;
      070222 001056          ;
10058          ;
10059 070224          540$:
10060 070224 012705 006472'   MOV    @RBUF+26.,R5  ;CHECK CRC
10061 070230 004737 030642'   JSR    PC,CMPCRC     ; ERRORS ?
10062 070234 103006          BCC    550$          ; NO
10063 070236          ERRHRD 503.,ERR023,MSG008 ; YES, REPORT ERROR
      070236 104456          ;
      070240 000767          ;
      070242 024417'         ;
      070244 022400'         ;
10064 070246          ESCAPE TST          ; AND ABORT TEST          TRAP C$ESCAPE
      070246 104410          ;
      070250 001030          ;
10065          ;
10066 070252          550$:
10067 070252 012777 004100 107744 MOV    @DNI!INTE,@PCSRO ; ENABLE INTERRUPTS
10068 070260 112777 000117 107736 MOVB   @INTE!STOP,@PCSRO ; ISSUE STOP PORT COMMAND
10069 070266 004737 026550'   JSR    PC,CHKDNI     ; DNI ?
10070 070272 103010          BCC    560$          ; YES
10071 070274          FTL
    
```

```

070274 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
10072 070300                ERRHRD  504.,ERR019,MSG003 ; NO, REPORT ERROR
      070300 104456                TRAP      C$ERHRD
      070302 000770                .WORD    504
      070304 024126'                .WORD    'R019
      070306 021736'                .WORD    'SG003
10073 070310                ESCAPE  TST                ; AND ABORT TEST
      070310 104410                TRAP      C$ESCAPE
      070312 000766                .WORD    L10054
10074
10075 070314 004737 030264' ;560$: JSR      PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
10076
10077 070320 103010          BCC      565$              ; ERROR ?
10078 070322                FTL                          ; NO
      070322 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
10079 070326                ERRHRD  505.,ERR006,MSG003 ; YES, REPORT ERROR
      070326 104456                TRAP      C$ERHRD
      070330 000771                .WORD    505
      070332 023030'                .WORD    ERR006
      070334 021736'                .WORD    MSG003
10080 070336                ESCAPE  TST                ; AND ABORT TEST
      070336 104410                TRAP      C$ESCAPE
      070340 000740                .WORD    L10054-.
10081
10082 070342                ;565$:
10083 070342 004737 030236'      JSR      PC,CLRCV          ; CLEAR RECEIVE BUFFER
10084 070346 062702 000006      ADD      #6,R2             ; UPDATE R2
10085 070352 062703 000004      ADD      #4,R3             ; UPDATE R3
10086 070356 005304            DEC      R4                 ; DONE TEN LOOPBACKS
10087 070360 001402            BEQ      566$              ; YES
10088 070362 000137 067520'      JMP      420$              ; NO
10089
10090
10091                ;DO TEN LOOPS WITH DEST ADDRESS = COMPLIMENTED MULTICAST ADDRESS
10092
10093 070366 012704 000012      ;566$: MOV      #10.,R4        ; DO LOOP = TEN
10094 070372 012702 016170'      MOV      #MULTLC,R2       ; R2 POINTS TO COMPLIMENTED LIST
10095
10096                ;SET UP RINGS FOR ONE BUFFER LOOPBACK
10097
10098 070376                ;570$:
10099 070376 012705 014410'      MOV      #TDRB1A,R5       ; DEFAULT ONE BUFFER TRANSMIT RING
10100 070402 004737 032164'      JSR      PC,LDTDRB        ; LOAD TDRB
10101 070406 012705 012750'      MOV      #RDRB1A,R5       ; DEFAULT ONE BUFFER RECEIVE RING
10102 070412 004737 032070'      JSR      PC,LDRDRB        ; LOAD RDRB
10103
10104                ;SET UP BUFFERS AND START
10105
10106 070416 012701 000264'      MOV      #SRC,R1          ; SOURCE = DEF PHYSICAL ADDR
10107 070422 004737 033206'      JSR      PC,SRCDST        ; DEST = COMPL MULTICAST ADDR
10108 070426 005037 016562'      CLR      D0CRC            ; NO APPEND CRC
10109 070432 012737 000006 016560' MOV      #6,BYTCNT        ; BYTES/PACKET
10110 070440 004737 033006'      JSR      PC,SETBUF        ; SET UP BUFFERS
10111 070444 012777 004100 107552 MOV      #DNI!INTE,@PCSRO ; ENABLE INTERRUPTS
  
```

10112	070452	112777	000104	107544	MOVB	@INTE!START,@PCSR0	; ISSUE START PORT COMMAND		
10113	070460	004737	026550'		JSR	PC,CHKDNI	; DNI?		
10114	070464	103010			BCC	580\$	; YES		
10115	070466				FTL				
	070466	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
10116	070472				ERRHRD	506.,ERR012,MSG003	; NO, REPORT ERROR		
	070472	104456						TRAP	C\$ERHRD
	070474	000772						.WORD	506
	070476	023447'						.WORD	ERR012
	070500	021736'						.WORD	MSG003
10117	070502				ESCAPE	TST	; AND ABORT TEST		
	070502	104410						TRAP	C\$ESCAPE
	070504	000574						.WORD	L10054 .
10118									
10119	070506	004737	030264'		JSR	PC,CLRDN1	; WRITE ONE TO CLEAR DNI		
10120							; ERROR ?		
10121	070512	103010			BCC	590\$	; NO		
10122	070514				FTL				
	070514	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
10123	070520				ERRHRD	507.,ERR006,MSG003	; YES, REPORT ERROR		
	070520	104456						TRAP	C\$ERHRD
	070522	000773						.WORD	507
	070524	023030'						.WORD	ERR006
	070526	021736'						.WORD	MSG003
10124	070530				ESCAPE	TST	; AND ABORT TEST		
	070530	104410						TRAP	C\$ESCAPE
	070532	000546						.WORD	L10054 .
10125									
10126	070534	004737	027566'		JSR	PC,CHKTXI	; TXI ?		
10127	070540	103010			BCC	600\$	; YES		
10128	070542				FTL				
	070542	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
10129	070546				ERRHRD	510.,ERR013,MSG003	; NO, REPORT ERROR		
	070546	104456						TRAP	C\$ERHRD
	070550	000776						.WORD	510
	070552	023530'						.WORD	ERR013
	070554	021736'						.WORD	MSG003
10130	070556				ESCAPE	TST	; AND ABORT TEST		
	070556	104410						TRAP	C\$ESCAPE
	070560	000520						.WORD	L10054 .
10131									
10132	070562	004737	030500'		JSR	PC,CLRTXI	; WRITE ONE TO CLEAR TXI		
10133							; ERROR ?		
10134	070566	103010			BCC	610\$	; NO		
10135	070570				FTL				
	070570	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
10136	070574				ERRHRD	511.,ERR014,MSG003	; YES, REPORT ERROR		
	070574	104456						TRAP	C\$ERHRD
	070576	000777						.WORD	511

	070600	023561'				.WORD	ERR014
	070602	021736'				.WORD	MSG003
10137	070604		ESCAPE	TST			
	070604	104410				TRAP	C\$ESCAPE
	070606	000472				.WORD	L10054
10138							
10139	070610	012705	000620'	610\$:	MOV	#TDRB,R5	
10140	070614	004737	027024'		JSR	PC,CHKOWN	
10141	070620	103010			BCC	620\$	
10142	070622				FTL		
	070622	004737	026652'		JSR	PC,CHKFTL	
10143	070626				ERRHRD	512.,ERR018	
	070626	104456					
	070630	001000					
	070632	024026'					
	070634	000000					
10144	070636		ESCAPE	TST			
	070636	104410					
	070640	000440					
10145							
10146	070642	012705	016264'	620\$:	MOV	#TDR14A,R5	
10147	070646	004737	032370'		JSR	PC,LDXTDR	
10148	070652	012705	000620'		MOV	#TDRB,R5	
10149	070656	004737	027500'		JSR	PC,CHKTDR	
10150	070662	103010			BCC	630\$	
10151	070664				FTL		
	070664	004737	026652'		JSR	PC,CHKFTL	
10152	070670				ERRHRD	513.,ERR020,MSG005	
	070670	104456					
	070672	001001					
	070674	024206'					
	070676	022042'					
10153	070700		ESCAPE	TST			
	070700	104410					
	070702	000376					
10154							
10155	070704	004737	027316'	630\$:	JSR	PC,CHKRXI	
10156	070710	103010			BCC	640\$	
10157	070712				FTL		
	070712	004737	026652'		JSR	PC,CHKFTL	
10158	070716				ERRHRD	514.,ERR015,MSG003	
	070716	104456					
	070720	001002					
	070722	023627'					
	070724	021736'					
10159	070726		ESCAPE	TST			
	070726	104410					
	070730	000350					
10160							
10161	070732	004737	030432'	640\$:	JSR	PC,CLRRI	
10162							



```

10163 070736 103010          BCC      650$          ; NO
10164 070740          FTL
      070740 004737 026652'   JSR      PC,CHKFTL    ; 'FATL' BIT SET?
10165 070744          ERRHRD   515.,ERR016,MSG003 ; YES, REPORT ERROR
      070744 104456          TRAP     C$ERHRD
      070746 001003          .WORD   515
      070750 023660'        .WORD   ERR016
      070752 021736'        .WORD   MSG003
10166 070754          ESCAPE   TST          ; AND ABORT TEST
      070754 104410          TRAP     C$ESCAPE
      070756 000322          .WORD   L10054-.
10167
10168 070760 012705 000660'   ; 650$: MOV     #RDRB,R5    ; CHECK RDRB OWNERSHIP
10169 070764 004737 027024'   JSR      PC,CHKOWN    ; OWN = PORT DRIVER ?
10170 070770 103010          BCC      660$          ; YES
10171 070772          FTL
      070772 004737 026652'   JSR      PC,CHKFTL    ; 'FATL' BIT SET?
10172 070776          ERRHRD   516.,ERR017    ; NO, REPORT ERROR
      070776 104456          TRAP     C$ERHRD
      071000 001004          .WORD   516
      071002 023726'        .WORD   ERR017
      071004 000000          .WORD   0
10173 071006          ESCAPE   TST          ; AND ABORT TEST
      071006 104410          TRAP     C$ESCAPE
      071010 000270          .WORD   L10054-.
10174
10175 071012 012705 016454'   ; 660$: MOV     #RDR20C,R5 ; POINT TO EXPECTED RDRB
10176 071016 012705 000660'   MOV     #RDRB,R5     ; CHECK RDRB
10177 071022 004737 027206'   JSR      PC,CHKRDR    ; ERRORS ?
10178 071026 103010          BCC      670$          ; NO
10179 071030          FTL
      071030 004737 026652'   JSR      PC,CHKFTL    ; 'FATL' BIT SET?
10180 071034          ERRHRD   517.,ERR021,MSG006 ; YES, REPORT ERROR
      071034 104456          TRAP     C$ERHRD
      071036 001005          .WORD   517
      071040 024267'        .WORD   ERR021
      071042 022204'        .WORD   MSG006
10181 071044          ESCAPE   TST          ; AND ABORT TEST
      071044 104410          TRAP     C$ESCAPE
      071046 000232          .WORD   L10054 .
10182
10183           ; COMPARE RBUF WITH TBUF
10184
10185 071050 013705 016560'   ; 670$: MOV     BYTCNT,R5 ; COMPARE DATA
10186 071054 004737 030712'   JSR      PC,CMPDAT    ; DATA COMPARE ERROR ?
10187 071060 103006          BCC      680$          ; NO
10188 071062          ERRHRD   520.,ERR022,MSG007 ; YES, REPORT ERROR
      071062 104456          TRAP     C$ERHRD
      071064 001010          .WORD   520
      071066 024350'        .WORD   ERR022
      071070 022346'        .WORD   MSG007
    
```

```

10189 071072          ESCAPE TST          . AND ABORT TEST
      071072 104410          TRAP          C$ESCAPE
      071074 000204          .WORD          L10054

10190
10191 071076          ;
      ; 680$:
10192 071076 012705 006472' MOV      #RBUF+26.,R5          ; CHECK CRC
10193 071102 004737 030642' JSR      PC,CMPCRC          ; ERRORS ?
10194 071106 103006          BCC      690$              ; NO
10195 071110          ERRHRD 521.,ERR023,MSG008 ; YES, REPORT ERROR
      071110 104456          TRAP          C$ERHRD
      071112 001011          .WORD          521
      071114 024417'          .WORD          ERR023
      071116 022400'          .WORD          MSG008

10196 071120          ESCAPE TST          ; AND ABORT TEST
      071120 104410          TRAP          C$ESCAPE
      071122 000156          .WORD          L10054 .

10197
10198 071124          ;
      ; 690$:
10199 071124 012777 004100 107072 MOV      #DNI!INTE,@PCSR0          ; ENABLE INTERRUPTS
10200 071132 112777 000117 107064 MOVB    #INTE!STOP,@PCSR0          ; ISSUE STOP PORT COMMAND
10201 071140 004737 026550' JSR      PC,CHKDNI          ; DNI ?
10202 071144 103010          BCC      700$              ; YES
10203 071146          FTL
      071146 004737 026652' JSR      PC,CHKFTL          ; 'FATL' BIT SET?
      ERRHRD 522.,ERR019,MSG003 ; NO, REPORT ERROR
      TRAP          C$ERHRD
      .WORD          522
      .WORD          ERR019
      .WORD          MSG003

10204 071152          ESCAPE TST          ; AND ABORT TEST
      071152 104456          TRAP          C$ESCAPE
      071154 001012          .WORD          L10054 .
      071156 024126'
      071160 021736'

10205 071162          ESCAPE TST          ; AND ABORT TEST
      071162 104410          TRAP          C$ESCAPE
      071164 000114          .WORD          L10054 .

10206
10207 071166 004737 030264' ;
      ; 700$:
10208          JSR      PC,CLRDNI          ; WHILE ONE TO CLEAR DNI
10209 071172 103010          BCC      710$              ; ERROR ?
10210 071174          FTL
      071174 004737 026652' JSR      PC,CHKFTL          ; 'FATL' BIT SET?
      ERRHRD 523.,ERR006,MSG003 ; YES, REPORT ERROR
      TRAP          C$ERHRD
      .WORD          523
      .WORD          ERR006
      .WORD          MSG003

10211 071200          ESCAPE TST          ; AND ABORT TEST
      071200 104456          TRAP          C$ESCAPE
      071202 001013          .WORD          L10054 .
      071204 023030'
      071206 021736'

10212 071210          ESCAPE TST          ; AND ABORT TEST
      071210 104410          TRAP          C$ESCAPE
      071212 000066          .WORD          L10054 .

10213
10214 071214          ;
      ; 710$:
10215 071214 004737 030236' JSR      PC,CLRCV          ; CLEAR RECEIVE BUFFER
10216 071220 062702 000006 ADD      #6,R2              ; UPDATE R2
10217 071224 062703 000004 ADD      #4,R3              ; UPDATE R3
10218 071230 005304          DEC      R4                ; DONE TEN LOOPBACKS
10219 071232 001402          BEQ      900$              ; YES
    
```



10234  
 10235  
 10236  
 10237  
 10238  
 10239  
 10240  
 10241  
 10242  
 10243  
 10244  
 10245  
 10246  
 10247  
 10248  
 10249  
 10250  
 10251  
 10252  
 10253  
 10254  
 10255  
 10256  
 10257  
 10258  
 10259  
 10260  
 10261  
 10262  
 10263  
 10264  
 10265  
 10266  
 10267  
 10268 071302  
 071302  
 10269  
 10270 071302  
 071302 012704 073534'  
 071306 004737 032734'  
 10271 071312 004737 033434'  
 10272 071316 103034  
 10273 071320 012777 004100 106676  
 10274 071326 112777 000140 106670  
 10275 071334 004737 027736'  
 10276 071340 103010  
 10277 071342  
 071342 004737 026652'  
 10278 071346  
 071346 104456  
 071350 001014

.SBTTL TEST 23: ENABLE ALL MULTICAST MODE TEST

```

*****
;
; THIS TEST VERIFIES THAT ENABLE ALL MULTICAST MODE
; IS OPERATIONAL.
; A WRITE MULTICAST ADDRESS LIST FUNCTION IS USED TO SET
; THE DELUA'S MULTICAST ADDRESS LIST.
; INTERNAL LOOPBACKS ARE THEN PERFORMED WITH
; CURRENTLY ENABLED AND THEN CURRENTLY DISABLED
; MULTICAST DESTINATION ADDRESSES.
; ALL LOOPBACKS ARE VERIFIED FOR SUCCESSFUL RECEPTION.
;
; TEST SEQUENCE:
; 1. WRITE MODE REGISTER = INTERNAL LOOPBACK
; and ENABLE ALL MULTICAST MODE
; 2. WRITE RING FORMAT
; 3. WRITE PHYSICAL ADDRESS
; 4. WRITE MULTICAST ADDRESS LIST
; 5. SET UP RINGS AND BUFFERS
; WITH DESTINATION ADDRESS = MULTICAST ADDRESS
; 6. ISSUE START
; 7. CHECK FOR ERRORS
; 8. ISSUE STOP
; 9. REPEAT STEPS 5 - 8 FOR ALL TEN LIST ENTRIES
; 10. SET UP RINGS AND BUFFERS
; WITH DESTINATION ADDRESS = COMPLIMENTED MULTICAST ADDRESS
; 11. ISSUE START
; 12. CHECK FOR ERRORS
; 13. ISSUE STOP
; 14. REPEAT STEPS 10 - 13 FOR ALL TEN ENTRIES
;
*****
    
```

BGNTST

T23::

PNTMAC T23ID

```

MOV    #T23ID,R4      ;GET POINTER TO TEST NAME MESSAGE
JSR    PC,PNTID       ;PRINT TEST NUMBER AND NAME
    
```

END OF MACRO EXPANSION OF 'PNTMAC'

```

JSR    PC,TINIT      ; IS A DEVICE RESET NEEDED?
BCC    30$           ; NO
MOV    #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
MOVB   #INTE!RSET,@PCSR0 ; YES, RESET DELUA
JSR    PC,CKDNI      ; DNI ?
BCC    20$           ; YES
FTL
    
```

```

JSR    PC,CHK^TL     ; 'FATL' BIT SET?
    
```

```

ERRHRD 524,.ERR006,MSG003 ; NO. REPORT ERROR
    
```

```

TRAP   C$ERHRD
.WORD  524
    
```

	071352	023030'					.WORD	ERR006
	071354	021736'					.WORD	MSG003
10279	071356		ESCAPE	TST				
	071356	104410					TRAP	C#ESCAPE
	071360	002216					.WORD	L10055-
10280								
10281	071362	004737	030264'	20#:	JSR	PC,CLRDNI		; WRITE ONE TO CLEAR DNI
10282								; ERROR ?
10283	071366	103010			BCC	30#		; NO
10284	071370				FTL			
	071370	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
10285	071374				ERRHRD	525.,ERR006,MSG003		; YES, REPORT ERROR
	071374	104456					TRAP	C#ERHRD
	071376	001015					.WORD	525
	071400	023030'					.WORD	ERR006
	071402	021736'					.WORD	MSG003
10286	071404		ESCAPE	TST				; AND ABORT TEST
	071404	104410					TRAP	C#ESCAPE
	071406	002170					.WORD	L10055-
10287								
10288	071410			30#:				
10289	071410	004737	030212'		JSR	PC,CLRBUF		; CLEAR RBUF AND TBUF
10290	071414	004737	031732'		JSR	PC,LDDFLT		; LOAD DEFAULT PHY.ADDRESS TABLES
10291	071420	004737	032032'		JSR	PC,LDPCSR		; ADDRESS OF PCBB -> PCSR2!3
10292	071424	012777	004100	106572	MOV	#DNI!INTE,@PCSR0		; ENABLE INTERRUPTS
10293	071432	112777	000101	106564	MOVB	#INTE!GETPCB,@PCSR0		; ISSUE GET_PCBB PORT COMMAND
10294	071440	004737	026550'		JSR	PC,CHKDNI		; DNI?
10295	071444	103010			BCC	40#		; YES
10296	071446				FTL			
	071446	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
10297	071452				ERRHRD	526.,ERR009,MSG003		; NO, REPORT ERROR
	071452	104456					TRAP	C#ERHRD
	071454	001016					.WORD	526
	071456	023245'					.WORD	ERR009
	071460	021736'					.WORD	MSG003
10298	071462		ESCAPE	TST				; AND ABORT TEST
	071462	104410					TRAP	C#ESCAPE
	071464	002112					.WORD	L10055-
10299								
10300	071466	004737	030264'	40#:	JSR	PC,CLRDNI		; WRITE ONE TO CLEAR DNI
10301								; ERROR ?
10302	071472	103010			BCC	50#		; NO
10303	071474				FTL			
	071474	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
10304	071500				ERRHRD	527.,ERR006,MSG003		; YES, REPORT ERROR
	071500	104456					TRAP	C#ERHRD
	071502	001017					.WORD	527
	071504	023030'					.WORD	ERR006
	071506	021736'					.WORD	MSG003
10305	071510		ESCAPE	TST				; AND ABORT TEST
	071510	104410					TRAP	C#ESCAPE

```

071512 002064 .WORD L10055-.
10306
10307 ;WRITE MODE REGISTER = INTERNAL LOOPBACK
10308 ; and ENABLF ALL MULTICAST MODE
10309 ;
10310 071514 012705 012600' 50$: MOV #WTMODE,R5 ; DEFAULT WRITE MODE FUNCTION
10311 071520 004737 032002' JSR PC,LDPCBB ; LOAD FUNCTION -> PCBB
10312 071524 013737 016474' 000302' MOV MODE24,PCBB+2 ; MODE = INTL LOOPBACK AND ENAL
10313 071532 012777 004100 106464 MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
10314 071540 112777 000102 106456 MOVB #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
10315 071546 004737 026550' JSR PC,CHKDNI ; DNI ?
10316 071552 103010 BCC 60$ ; YES
10317 071554 FTL

071554 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

10318 071560 ERRHRD 530.,ERR010,MSG003 ; NO, REPORT ERROR
071560 104456 TRAP C#ERHRD
071562 001022 .WORD 530
071564 023331' .WORD ERR010
071566 021736' .WORD MSG003

10319 071570 ESCAPE TST ; AND ABORT TEST
071570 104410 TRAP C#ESCAPE
071572 002004 .WORD L10055-.

10320
10321 071574 004737 030264' 60$: JSR PC,CLRDNI ; WRITE ONE TO CLEAR DNI
10322 BCC 70$ ; ERROR ?
10323 071600 103010 FTL ; NO
10324 071602

071602 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

10325 071606 ERRHRD 531.,ERR006,MSG003 ; YES, REPORT ERROR
071606 104456 TRAP C#ERHRD
071610 001023 .WORD 531
071612 023030' .WORD ERR006
071614 021736' .WORD MSG003

10326 071616 ESCAPE TST ; AND ABORT TEST
071616 104410 TRAP C#ESCAPE
071620 001756 .WORD L10055-.

10327
10328 ;WRTE RING FORMAT
10329 071622 012705 01254' 70$: MOV #WTRNGS,R5 ; DEFAULT WRITE RING FORMAT FUNCTION
10330 071626 004737 032002' JSR PC,LDPCBB ; LOAD FUNCTION -> PCBB
10331 071632 012705 012704' MOV #RFRMT,R5 ; DEFAULT RING FORMAT
10332 071636 012700 000006 MOV #6,R0 ; FORMAT = SIX WORDS
10333 071642 004737 032260' JSR PC,LDUDBB ; LOAD RING FORMAT -> UDBB
10334 071646 012777 004100 106350 MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
10335 071654 112777 000102 106342 MOVB #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
10336 071662 004737 026550' JSR PC,CHKDNI ; DNI ?
10337 071666 103010 BCC 80$ ; YES
10338 071670 FTL

071670 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

10339 071674 ERRHRD 532.,ERR010,MSG003 ; NO, REPORT ERROR
071674 104456 TRAP C#ERHRD
    
```

```

    071676 001024          .WORD 532
    071700 023331'       .WORD ERR010
    071702 021736'       .WORD MSG003
10340 071704          ESCAPE TST          ; AND ABORT TEST
    071704 104410          TRAP C$ESCAPE
    071706 001670          .WORD L10055-.
10341
10342 071710 004737 030264' ; 80$: JSR PC,CLRDNI          ; WRITE ONE TO CLEAR DNI
10343                                     ; ERROR ?
10344 071714 103010          BCC 90$          ; NO
10345 071716          FTL
                                     ;
    071716 004737 026652'   JSR PC,CHKFTL          ; 'FATL' BIT SET?
10346 071722          ERRHRD 533.,ERR006,MSG003 ; YES, REPORT ERROR
    071722 104456          TRAP C$ERHRD
    071724 001025          .WORD 533
    071726 023030'       .WORD ERF006
    071730 021736'       .WORD MSG003
10347 071732          ESCAPE TST          ; AND ABORT TEST
    071732 104410          TRAP C$ESCAPE
    071734 001642          .WORD L10055-.
10348
10349                                     ; WRITE PHYSICAL ADDRESS
10350
10351 071736          90$:
10352 071736 012705 000272'   MOV $DFault,R5          ; POINT TO DEFAULT PHYS. ADDRESS
10353 071742 004737 032050'   JSR PC,LDPHYA          ; SAVE DEF PHY ADDR IN DEF TABLE
10354 071746 012705 012500'   MOV $WTPHYA,R5        ; DEFAULT WRITE PHYSICAL ADDR FUNC
10355 071752 004737 032002'   JSR PC,LDPCCBB        ; LOAD FUNCTION > PCBB
10356 071756 012777 004100 106240 MOV $DNI!INTE,@PCSR0   ; ENABLE INTERRUPTS
10357 071764 112777 000102 106232 MOVB $INTE!GETCMD,@PCSR0 ; ISSUE GET CMD PORT COMMAND
10358 071772 004737 026550'   JSR PC,CHKDNI          ; DNI ?
10359 071776 103010          BCC 100$          ; YES
10360 072000          FTL
                                     ;
    072000 004737 026652'   JSR PC,CHKFTL          ; 'FATL' BIT SET?
10361 072004          ERRHRD 534.,ERR010,MSG003 ; NO, REPORT ERROR
    072004 104456          TRAP C$ERHRD
    072006 001026          .WORD 534
    072010 023331'       .WORD ERR010
    072012 021736'       .WORD MSG003
10362 072014          ESCAPE TST          ; AND ABORT TEST
    072014 104410          TRAP C$ESCAPE
    072016 001560          .WORD L10055-.
10363
10364 072020 004737 030264' ; 100$: JSR PC,CLRDNI          ; WRITE ONE TO CLEAR DNI
10365                                     ; ERROR ?
10366 072024 103010          BCC 102$          ; NO
10367 072026          FTL
                                     ;
    072026 004737 026652'   JSR PC,CHKFTL          ; 'FATL' BIT SET?
10368 072032          ERRHRD 535.,ERR006,MSG003 ; YES, REPORT ERROR
    072032 104456          TRAP C$ERHRD
    072034 001027          .WORD 535

```

```

072036 023030' .WORD ERR006
072040 021736' .WORD MSG003
10369 072042 ESCAPE TST ; AND ABORT TEST
072042 104410 TRAP C$ESCAPE
072044 001532 .WORD L10055-.

10370 ;
10371 ;WRITE MULTICAST ADDRESS LIST
10372 ;
10373 072046 012705 012520' 102$: MOV #WTMULA,R5 ; DEFAULT WRITE MULTICAST ADDR FUNC
10374 072052 004737 032002' JSR PC,LDP CBB ; LOAD FUNCTION -> PCBB
10375 072056 012705 016074' MOV #MULTL,R5 ; LOAD LIST INTO UDBB
10376 072062 012700 000036 MOV #30.,R0 ; LOAD 30 ENTRIES
10377 072066 004737 032260' JSR PC,LDUDBB ; MULTICAST LIST -> UDBB
10378 072072 012777 004100 106124 MOV #DNI!INTE,@PCSRO ; ENABLE INTERRUPTS
10379 072100 112777 000102 106116 MOVB #INTE!GETCMD,@PCSRO ; ISSUE GET_CMD PORT COMMAND
10380 072106 004737 026550' JSR PC,CHKDNT ; DNI ?
10381 072112 103010 BCC 104$ ; YES
10382 072114 FTL

072114 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

10383 072120 ERRHRD 536.,ERR010,MSG003 ; NO, REPORT ERROR
072120 104456 TRAP C$ERHRD
072122 001030 .WORD 536
072124 023331' .WORD ERR010
072126 021736' .WORD MSG003
10384 072130 ESCAPE TST ; AND ABORT TEST
072130 104410 TRAP C$ESCAPE
072132 001444 .WORD L10055-.

10385 ;
10386 072134 004737 030264' 104$: JSR PC,CLR DNI ; WRITE ONE TO CLEAR DNI
10387 BCC 106$ ; ERROR ?
10388 072140 103010 FTL ; NO
10389 072142 FTL

072142 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

10390 072146 ERRHRD 537.,ERR006,MSG003 ; YES, REPORT ERROR
072146 104456 TRAP C$ERHRD
072150 001031 .WORD 537
072152 023030' .WORD ERR006
072154 021736' .WORD MSG003
10391 072156 ESCAPE TST ; AND ABORT TEST
072156 104410 TRAP C$ESCAPE
072160 001416 .WORD L10055 .

10392 ;
10393 ;DO TEN LOOPS WITH DEST ADDR = MULTICAST ADDRESS
10394 ;
10395 072162 106$: JSR PC,CLRCV ; CLEAR RECEIVE BUFFER
10396 072162 004737 030236' MOV #10.,R4 ; DO LOOP = TEN
10397 072166 012704 000012 MOV #MULTL,R2 ; R2 POINTS TO MULTICAST LIST
10398 072172 012702 016074'

10399 ;
10400 ;SET UP RINGS FOR ONE BUFFER LOOPBACK
10401 ;
10402 072176 012705 014410' 110$: MOV #TDRB1A,R5 ; DEFAULT ONE BUFFER TRANSMIT RING
10403 072202 004737 032164' JSR PC,LDTDRB ; LOAD TDRB
    
```



```

10404 072206 012705 012750'      MOV    #RDRB1A,R5      ; DEFAULT ONE BUFFER RECEIVE RING
10405 072212 004737 032070'      JSR    PC,LDRDRB      ; LOAD RDRB
10406                                     ;
10407                                     ;SET UP BUFFERS AND START
10408
10409 072216 012701 000264'      MOV    #SRC,R1        ; SOURCE = PHYSICAL ADDRESS
10410 072222 004737 033206'      JSR    PC,SRCDST      ; DEST = MULTICAST ADDRESS
10411 072226 005037 016562'      CLR    D0CRC          ; NO APPEND CRC
10412 072232 012737 000006 016560'  MOV    #6,BYTCNT      ; BYTES/PACKET
10413 072240 004737 033006'      JSR    PC,SETBUF      ; SET UP BUFFERS
10414 072244 012777 004100 105752  MOV    #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
10415 072252 112777 000104 105744  MOVB   #INTE!START,@PCSR0 ; ISSUE START PORT COMMAND
10416 072260 004737 026550'      JSR    PC,CHKDNI      ; DNI?
10417 072264 103010                BCC    120$           ; YES
10418 072266
                                     ;
                                     JSR    PC,CHKFTL      ; 'FATL' BIT SET?
072266 004737 026652'
10419 072272                ERRHRD  540.,ERR012,MSG003 ; NO, REPORT ERROR
072272                TRAP    C$ERHRD
072274 104456                .WORD  540
072276 023447'                .WORD  ERR012
072300 021736'                .WORD  MSG003
10420 072302                ESCAPE  TST           ; AND ABORT TEST
072302 104410                TRAP    C$ESCAPE
072304 001272                .WORD  L10055 .
10421
10422 072306 004737 030264'      ; 120$: JSR    PC,CLRDN1      ; WRITE ONE TO CLEAR DNI
10423                                     ; ERROR ?
10424 072312 103010                BCC    130$           ; NO
10425 072314
                                     ;
                                     JSR    PC,CHKFTL      ; 'FATL' BIT SET?
072314 004737 026652'
10426 072320                ERRHRD  541.,ERR006,MSG003 ; YES, REPORT ERROR
072320                TRAP    C$ERHRD
072322 104456                .WORD  541
072324 023030'                .WORD  ERR006
072326 021736'                .WORD  MSG003
10427 072330                ESCAPE  TST           ; AND ABORT TEST
072330 104410                TRAP    C$ESCAPE
072332 001244                .WORD  L10055 .
10428
10429 072334 004737 027566'      ; 130$: JSR    PC,CHKTXI      ; TXI ?
10430 072340 103010                BCC    140$           ; YES
10431 072342
                                     ;
                                     JSR    PC,CHKFTL      ; 'FATL' BIT SET?
072342 004737 026652'
10432 072346                ERRHRD  542.,ERR013,MSG003 ; NO, REPORT ERROR
072346                TRAP    C$ERHRD
072350 104456                .WORD  542
072352 023530'                .WORD  ERR013
072354 021736'                .WORD  MSG003
10433 072356                ESCAPE  TST           ; AND ABORT TEST
072356 104410                TRAP    C$ESCAPE
072360 001216                .WORD  L10055-.
    
```

10434									
10435	072362	004737	030500'	140\$:	JSR	PC,CLRTXI			; WRITE ONE TO CLEAR TXI
10436									; ERROR ?
10437	072366	103010			BCC	150\$			; NO
10438	072370				FTL				
	072370	004737	026652'		JSR	PC,CHKFTL			; 'FATL' BIT SET?
10439	072374				ERRHRD	543.,ERR014,MSG003			; YES, REPORT ERROR
	072374	104456						TRAP	C\$ERHRD
	072376	001037						.WORD	543
	072400	023561'						.WORD	ERR014
	072402	021736'						.WORD	MSG003
10440	072404				ESCAPE	TST			; AND ABORT TEST
	072404	104410						TRAP	C\$ESCAPE
	072406	001170						.WORD	L10055-.
10441									
10442	072410	012705	000620'	150\$:	MOV	#TDRB,R5			; CHECK TDRB OWNERSHIP
10443	072414	004737	027024'		JSR	PC,CHKOWN			; OWN = PORT DRIVER ?
10444	072420	103010			BCC	160\$			; YES
10445	072422				FTL				
	072422	004737	026652'		JSR	PC,CHKFTL			; 'FATL' BIT SET?
10446	072426				ERRHRD	544.,ERR018			; NO, REPORT ERROR
	072426	104456						TRAP	C\$ERHRD
	072430	001040						.WORD	544
	072432	024026'						.WORD	ERR018
	072434	000000						.WORD	0
10447	072436				ESCAPE	TST			; AND ABORT TEST
	072436	104410						TRAP	C\$ESCAPE
	072440	001136						.WORD	L10055-.
10448									
10449	072442	012705	016264'	160\$:	MOV	#TDR14A,R5			; POINT TO EXPECTED TDRB
10450	072446	004737	032370'		JSR	PC,LDXTDR			; LOAD INTO XTDRBO TABLE
10451	072452	012705	000620'		MOV	#TDRB,R5			; CHECK TDRB
10452	072456	004737	027500'		JSR	PC,CHKTDR			; ERRORS ?
10453	072462	103010			BCC	170\$			; NO
10454	072464				FTL				
	072464	004737	026652'		JSR	PC,CHKFTL			; 'FATL' BIT SET?
10455	072470				ERRHRD	545.,ERR020,MSG005			; YES, REPORT EPROR
	072470	104456						TRAP	C\$ERHRD
	072472	001041						.WORD	545
	072474	024206'						.WORD	ERR020
	072476	022042'						.WORD	MSG005
10456	072500				ESCAPE	TST			; AND ABORT TEST
	072500	104410						TRAP	C\$ESCAPE
	072502	001074						.WORD	L10055-.
10457									
10458	072504	004737	027316'	170\$:	JSR	PC,CHKRXI			; RXI ?
10459	072510	103010			BCC	180\$			; YES
10460	072512				FTL				
	072512	004737	026652'		JSR	PC,CHKFTL			; 'FATL' BIT SET?

10461	072516			ERRHRD	546.,ERR015,MSG003	; NO, REPORT ERROR		
	072516	104456					TRAP	C\$ERHRD
	072520	001042					.WORD	546
	072522	023627'					.WORD	ERR015
	072524	021736'					.WORD	MSG003
10462	072526			ESCAPE	TST	; AND ABORT TEST		
	072526	104410					TRAP	C\$ESCAPE
	072530	001046					.WORD	L10055-.
10463								
10464	072532	004737	030432'	180\$:	JSR	PC,CLRRXI		; WRITE ONE TO CLEAR RXI
10465								; ERROR ?
10466	072536	103010			BCC	190\$		; NO
10467	072540				FTL			
	072540	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
10468	072544			ERRHRD	547.,ERR016,MSG003	; YES, REPORT ERROR		
	072544	104456					TRAP	C\$ERHRD
	072546	001043					.WORD	547
	072550	023660'					.WORD	ERR016
	072552	021736'					.WORD	MSG003
10469	072554			ESCAPE	TST	; AND ABORT TEST		
	072554	104410					TRAP	C\$ESCAPE
	072556	001020					.WORD	L10055-.
10470								
10471	072560	012705	000660'	190\$:	MOV	#RDRB,R5		; CHECK RDRB OWNERSHIP
10472	072564	004737	027024'		JSR	PC,CHKOWN		; OWN = PORT DRIVER ?
10473	072570	103010			BCC	200\$		; YES
10474	072572				FTL			
	072572	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
10475	072576			ERRHRD	550.,ERR017	; NO, REPORT ERROR		
	072576	104456					TRAP	C\$ERHRD
	072600	001046					.WORD	550
	072602	023726'					.WORD	ERR017
	072604	000000					.WORD	0
10476	072606			ESCAPE	TST	; AND ABORT TEST		
	072606	104410					TRAP	C\$ESCAPE
	072610	000766					.WORD	L10055-.
10477								
10478	072612	012705	016454'	200\$:	MOV	#RDR20C,R5		; POINT TO EXPECTED RDRB
10479	072616	004737	032340'		JSR	PC,LDXRDR		; LOAD INTO XRDRBO TABLE
10480	072622	012705	000660'		MOV	#RDRB,R5		; CHECK RDRB
10481	072626	004737	027206'		JSR	PC,CHKRDR		; ERRORS ?
10482	072632	103010			BCC	210\$		; NO
10483	072634				FTL			
	072634	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
10484	072640			ERRHRD	551.,ERR021,MSG006	; YES, REPORT ERROR		
	072640	104456					TRAP	C\$ERHRD
	072642	001047					.WORD	551
	072644	024267'					.WORD	ERR021
	072646	022204'					.WORD	MSG006
10485	072650			ESCAPE	TST	; AND ABORT TEST		
	072650	104410					TRAP	C\$ESCAPE

```

072652 000724                                .WORD  L10055 .
10486
10487      ; COMPARE RBUF WITH TBUF
10488
10489 072654 013705 016560'      210$:  MOV     BYTCNT,R5      ; COMPARE DATA
10490 072660 004737 030712'      JSR     PC,CMPDAT      ; DATA COMPARE ERROR ?
10491 072664 103006              BCC     220$           ; NO
10492 072666              ERRHRD  552.,ERR022,MSG007 ; YES, REPORT ERROR
      072666 104456              TRAP   C$ERHRD
      072670 001050              .WORD  552
      072672 024350'              .WORD  ERR022
      072674 022346'              .WORD  MSG007
10493 072676              ESCAPE  TST           ; AND ABORT TEST
      072676 104410              TRAP   C$ESCAPE
      072700 000676              .WORD  L10055-.
10494
10495 072702      ;
10496 072702 012705 006472'      220$:  MOV     @RBUF+26.,R5    ; CHECK CRC
10497 072706 004737 030642'      JSR     PC,CMPDRC      ; ERRORS ?
10498 072712 103006              BCC     230$           ; NO
10499 072714              ERRHRD  553.,ERR023,MSG008 ; YES, REPORT ERROR
      072714 104456              TRAP   C$ERHRD
      072716 001051              .WORD  553
      072720 024417'              .WORD  ERR023
      072722 022400'              .WORD  MSG008
10500 072724              ESCAPE  TST           ; AND ABORT TEST
      072724 104410              TRAP   C$ESCAPE
      072726 000650              .WORD  L10055-.
10501
10502 072730      ;
10503 072730 012777 004100 105266 230$:  MOV     @DNI!INTE,@PCSR0  ; ENABLE INTERRUPTS
10504 072736 112777 000117 105260  MOVB   @INTE!STOP,@PCSR0 ; ISSUE STOP PORT COMP,AND
10505 072744 004737 026550'      JSR     PC,CHKDNI      ; DNI ?
10506 072750 103010              BCC     240$           ; YES
10507 072752              FTL
      072752 004737 026652'      JSR     PC,CHKFTL      ; 'FATL' BIT SET?
10508 072756              ERRHRD  554.,ERR019,MSG003 ; NO, REPORT ERROR
      072756 104456              TRAP   C$ERHRD
      072760 001052              .WORD  554
      072762 024126'              .WORD  ERR019
      072764 021736'              .WORD  MSG003
10509 072766              ESCAPE  TST           ; AND ABORT TEST
      072766 104410              TRAP   C$ESCAPE
      072770 000606              .WORD  L10055 .
10510
10511 072772 004737 030264'      240$:  JSR     PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
10512              BCC     245$           ; ERROR ?
10513 072776 103010              BCC     245$           ; NO
10514 073000              FTL
      073000 004737 026652'      JSR     PC,CHKFTL      ; 'FATL' BIT SET?
10515 073004              ERRHRD  555.,ERR006,MSG003 ; YES, REPORT ERROR
      073004 104456              TRAP   C$ERHRD
      073006 001053              .WORD  555
  
```

```

073010 023030' .WORD ERR006
073012 021736' .WORD MSG003
10516 073014 ESCAPE TST ; AND ABORT TEST TRAP C$ESCAPE
073014 104410 .WORD L10055-
073016 000560

10517 ;
10518 073020 ;245$:
10519 073020 004737 030236' JSR PC,CLRCV ; CLEAR RECEIVE BUFFER
10520 073024 062702 000006 ADD #6,R2 ; UPDATE R2
10521 073030 062703 000004 ADD #4,R3 ; UPDATE R3
10522 073034 005304 DEC R4 ; DONE TEN LOOPBACKS
10523 073036 001402 BEQ 246$ ; YES
10524 073040 000137 072176' JMP 110$ ; NO
10525 ;
10526 ;
10527 ;DO TEN LOOPS WITH DEST ADDRESS = COMPLIMENTED MULTICAST ADDRESS
10528 ;
10529 073044 012704 000012 ;246$: MOV #10.,R4 ; DO LOOP = TEN
10530 073050 012702 016170' MOV #MULTLC,R2 ; R2 POINTS TO COMPLIMENTED LIST
10531 ;
10532 ;SET UP RINGS FOR ONE BUFFER LOOPBACK
10533 ;
10534 073054 012705 014410' ;250$: MOV #TDRB1A,R5 ; DEFAULT ONE BUFFER TRANSMIT RING
10535 073060 004737 032164' JSR PC,LDTDRB ; LOAD TDRB
10536 073064 012705 012750' MOV #RDRB1A,R5 ; DEFAULT ONE BUFFER RECEIVE RING
10537 073070 004737 032070' JSR PC,LDRDRB ; LOAD RDRB
10538 ;
10539 ;SET UP BUFFERS AND START
10540 ;
10541 073074 012701 000264' MOV #SRC,R1 ; SOURCE = PHYSICAL ADDRESS
10542 073100 004737 033206' JSR PC,SRCDST ; DEST = COMPL MULTICAST ADDR
10543 073104 005037 016562' CLR D0CRC ; NO APPEND CRC
10544 073110 012737 000006 016560' MOV #6,BYTCNT ; BYTES/PACKET
10545 073116 004737 033006' JSR PC,SETBUF ; SET UP BUFFERS
10546 073122 012777 004100 105074 MOV #DNI!INTE,@PCSRO ; ENABLE INTERRUPTS
10547 073130 112777 000104 105066 MOVB #INTE!START,@PCSRO ; ISSUE START PORT COMMAND
10548 073136 004737 026550' JSR PC,CHKDNI ; DNI?
10549 073142 103010 BCC 260$ ; YES
10550 073144 FTL

073144 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

10551 073150 ERRHRD 556.,ERR012,MSG003 ; NO, REPORT ERROR
073150 104456 TRAP C$ERHRD
073152 001054 .WORD 556
073154 023447' .WORD ERR012
073156 021736' .WORD MSG003
10552 073160 ESCAPE TST ; AND ABORT TEST TRAP C$ESCAPE
073160 104410 .WORD L10055-
073162 000414

10553 ;
10554 073164 004737 030264' ;260$: JSR PC,CLRDN1 ; WRITE ONE TO CLEAR DNI
10555 ; ERROR ?
10556 073170 103010 BCC 270$ ; NO
10557 073172 FTL

073172 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?
    
```



```

10581 073320 012705 016264'      300$:  MOV    #TDR14A,R5      ; POINT TO EXPECTED TDRB
10582 073324 004737 032370'      JSR    PC,LDXTDR      ; LOAD INTO XTDRGO TAB F
10583 073330 012705 000620'      MOV    #TDRB,R5      ; CHECK TDRB
10584 073334 004737 027500'      JSR    PC,CHKTDR      ; ERRORS ?
10585 073340 103010                BCC    420$           ; NO
10586 073342                FTL

      073342 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?
10587 073346                ERRHRD 563.,ERR020,MSG005 ; YES, REPORT ERROR
      073346 104456                TRAP  C$ERHRD
      073350 001063                .WORD 563
      073352 024206'                .WORD ERR020
      073354 022042'                .WORD MSG005
10588 073356                ESCAPE TST           ; AND ABORT TEST
      073356 104410                TRAP  C$ESCAPE
      073360 000216                .WORD L10055-.
10589                ;
10590 073362 004737 032422'      ; 420$: JSR    PC,NORXI      ; RXI ?
10591 073366 103010                BCC    480$           ; NO
10592 073370                FTL

      073370 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?
10593 073374                ERRHRD 564.,ERR039,MSG003 ; YES, REPORT ERROR
      073374 104456                TRAP  C$ERHRD
      073376 001064                .WORD 564
      073400 025575'                .WORD ERR039
      073402 021736'                .WORD MSG003
10594 073404                ESCAPE TST           ; AND ABORT TEST
      073404 104410                TRAP  C$ESCAPE
      073406 000170                .WORD L10055 .
10595                ;
10596 073410                ; 480$:
10597 073410 012777 004100 104606  MOV    #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
10598 073416 112777 000117 104600  MOVB   #INTE!STOP,@PCSR0 ; ISSUE STOP PORT COMMAND
10599 073424 004737 026550'      JSR    PC,CHKDNI      ; DNI ?
10600 073430 103010                BCC    490$           ; YES
10601 073432                FTL

      073432 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?
10602 073436                ERRHRD 565.,ERR019,MSG003 ; NO, REPORT ERROR
      073436 104456                TRAP  C$ERHRD
      073440 001065                .WORD 565
      073442 024126'                .WORD ERR019
      073444 021736'                .WORD MSG003
10603 073446                ESCAPE TST           ; AND ABORT TEST
      073446 104410                TRAP  C$ESCAPE
      073450 000126                .WORD L10055-.
10604                ;
10605 073452 004737 030264'      ; 490$: JSR    PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
10606                ; ERROR ?
10607 073456 103010                BCC    495$           ; NO
10608 073460                FTL

      073460 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?
    
```

```

10609 073464          ERRPRD  566.,ERR006,MSG003      ; YES, REPORT ERROR
      073464 104456
      073466 001066
      073470 023030'
      073472 021736'
10610 073474          ESCAPE  TST                    ; AND ABORT TEST
      073474 104410
      073476 000100
10611
10612 073500          ;
      073500          ;495$:
10613 073500 004737 030236'      JSR      PC,CLRCV      ; CLEAR RECEIVE BUFFER
10614 073504 062702 000006      ADD      #6,R2        ; UPDATE R2
10615 073510 062703 000004      ADD      #4,R3        ; UPDATE R3
10616 073514 005304          DEC      R4           ; DONE TEN LOOPBACKS ?
10617 073516 001402          BEQ      500$          ; YES
10618 073520 000137 073054'      JMP      250$          ; NO
10619
10620 073524          ;
      073524          ;500$:
10621 073524 004737 030346'      JSR      PC,CLINTR    ; INSURE DELUA INTR BITS CLEAR
10622
10623 073530          EXIT      TST
      073530 104432
      073532 000044
10624
10625          ;LOCAL TEST MESSAGE
10626
10627 073534          104      105      114      T23ID: .ASCIZ 'DELUA ENABLE ALL MULTICAST MODE '
      073537          125      101      040
      073542          105      116      101
      073545          102      114      105
      073550          040      101      114
      073553          114      040      115
      073556          125      114      124
      073561          111      103      101
      073564          123      124      040
      073567          115      117      104
      073572          105      040      000
10628          .EVEN
10629
10630 073576          ENDTST
      073576
      073576 104401
      L10055:
      TRAP      C$ETST
  
```



10632  
 10633  
 10634  
 10635  
 10636  
 10637  
 10638  
 10639  
 10640  
 10641  
 10642  
 10643  
 10644  
 10645  
 10646  
 10647  
 10648  
 10649  
 10650  
 10651 073600  
       073600  
 10652  
 10653 073600  
       073600 012704 075162'  
       073604 004737 032734'  
 10654 073610 004737 033434'  
 10655 073614 103034  
 10656 073616 012777 004100 104400  
 10657 073624 112777 000140 104372  
 10658 073632 004737 027736'  
 10659 073636 103010  
 10660 073640  
       073640 004737 026652'  
 10661 073644  
       073644 104456  
       073646 001067  
       073650 023030'  
       073652 021736'  
 10662 073654  
       073654 104410  
       073656 001364  
 10663  
 10664 073660 004737 030264' 20\$:  
 10665  
 10666 073664 103010  
 10667 073666  
       073666 004737 026652'  
 10668 073672  
       073672 104456

.SBTTL TEST 24: INTERNAL LOOPBACK TRANSMIT LENGTH ERROR TEST

```

*****
:
: THIS TEST VERIFIES THAT, IF THE PORT DRIVER ATTEMPTS TO
: TRANSMIT GREATER THAN 32 BYTE <DTCR = 0>, OR 36 BYTE <DTCR = 1>
: TRANSMIT FRAME, THE DEVICE WILL RETURN A 'TRANSMIT LENGTH' ERROR.
:
: TEST SEQUENCE:
:   1. WRITE MODE REGISTER = INTERNAL LOOPBACK, PROM MODE
:   2. WRITE RING FORMAT
:   3. WRITE PHYSICAL ADDRESS
:   4. SET UP RINGS AND BUFFERS
:   5. ISSUE START
:   6. CHECK FOR BUFFER LENGTH ERROR IN TDRB+6
:
*****
    
```

BGNTST

T24::

PNTMAC T24ID

```

MOV   #T24ID,R4      ;GET POINTER TO TEST NAME MESSAGE
JSR   PC,PNTID       ;PRINT TEST NUMBER AND NAME
    
```

; END OF MACRO EXPANSION OF 'PNTMAC'

```

JSR   PC,TINIT       ; IS A DEVICE RESET NEEDED?
BCC   30$            ; NO
MOV   #DNI!INTE,@PCSR0 ;ENABLE INTERRUPTS
MOV   #INTE!RSET,@PCSR0 ; YES, RESET DELUA
JSR   PC,CKDNI       ; DNI ?
BCC   20$            ; YES
FTL
    
```

```

JSR   PC,CHKFTL      ; 'FATL' BIT SET?
    
```

```

ERRHRD 567.,ERR006,MSG003 ; NO, REPORT ERROR
    
```

```

TRAP   C$ERHRD
.WORD  567
.WORD  ERR006
.WORD  MSG003
    
```

```

ESCAPE TST           ; AND ABORT TEST
    
```

```

TRAP   C$ESCAPE
.WORD  L10056-.
    
```

```

20$: JSR   PC,CLRDN1  ; WRITE ONE TO CLEAR DNI
      BCC   30$       ; ERROR ?
      FTL           ; NO
    
```

```

JSR   PC,CHKFTL      ; 'FATL' BIT SET?
    
```

```

ERRHRD 570.,ERR006,MSG003 ; YES, REPORT ERROR
    
```

```

TRAP   C$ERHRD
    
```

```

073674 001072 .WORD 570
073676 023030' .WORD ERR006
073700 021736' .WORD MSG003
10669 073702 ESCAPE TST ; AND ABORT TEST
073702 104410 TRAP C$ESCAPE
073704 001336 .WORD L10056-.

10670 ;
10671 073706 ; 30$:
10672 073706 004737 031732' JSR PC,LDDFLT ; LOAD DEFAULT PHY. ADDRESS TABLES
10673 073712 012704 000001 MOV #' ,R4 ; INIT PASS COUNTER
10674 073716 ;
10675 073716 004737 030212' JSR PC,CLRBUF ; CLEAR TBUF AND RBUF
10676 073722 004737 032032' JSR PC,LDPCSR ; ADDRESS OF PCBB -> PCSR2!3
10677 073726 012777 004100 104270 MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
10678 073734 112777 000101 104262 MOVB #INTE!GETPCB,@PCSR0 ; ISSUE GET_PCBB PORT COMMAND
10679 073742 004737 026550' JSR PC,CHKDNI ; DNI?
10680 073746 103010 BCC 40$ ; YES
10681 073750 FTL

073750 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

10682 073754 ERRHRD 571.,ERR009,MSG003 ; NO, REPORT ERROR
073754 104456 TRAP C$ERHRD
073756 001073 .WORD 571
073760 023245' .WORD ERR009
073762 021736' .WORD MSG003
10683 073764 ESCAPE TST ; AND ABORT TEST
073764 104410 TRAP C$ESCAPE
073766 001254 .WORD L10056-.

10684 ;
10685 073770 004737 030264' ; 40$: JSR PC,CLR DNI ; WRITE ONE TO CLEAR DNI
10686 ; ; ERROR ?
10687 073774 103010 BCC 50$ ; NO
10688 073776 FTL

073776 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

10689 074002 ERRHRD 572.,ERR006,MSG003 ; YES, REPORT ERROR
074002 104456 TRAP C$ERHRD
074004 001074 .WORD 572
074006 023030' .WORD ERR006
074010 021736' .WORD MSG003
10690 074012 ESCAPE TST ; AND ABORT TEST
074012 104410 TRAP C$ESCAPE
074014 001226 .WORD L10056-.

10691 074016 ; 50$:
10692 ;
10693 ;WRITE MODE REGISTER = INTERNAL LOOPBACK AND PROM MODE
10694 ;
10695 ;IF 2ND PASS SKIP 1ST PASS PCBB SETUP
10696 ;
10697 074016 005304 DEC R4 ;ADJUST PASS COUNT
10698 074020 005704 TST R4 ;1ST PASS
10699 074022 100425 BMI 55$ ;NO, SKIP THIS SETUP
10700 074024 012705 012600' MOV #WTHMODE,R5 ; WRITE MODE FUNCTION (NO CRC)
10701 074030 004737 032002' JSR PC,LDPCBB ; LOAD FUNCTION -> PCBB
10702 074034 012777 004100 104162 MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS

```

```

10703 074042 112777 000102 104154      MOVB    #INTE!GETCMD,@PCSR0      ; ISSUE GET_CMD PORT COMMAND
10704 074050 004737 026550'          JSR     PC,CHKDNI                ; DNI ?
10705 074054 103035                    BCC     60$                      ; YES
10706 074056                    FTL

      074056 004737 026652'          JSR     PC,CHKFTL                ; 'FATL' BIT SET?

10707 074062                    ERRHRD  573.,ERR010,MSG003        ; NO, REPORT ERROR
      074062 104456                    TRAP   C,ERHRD
      074064 001075                    .WORD  573
      074066 023331'                    .WORD  ERR010
      074070 021736'                    .WORD  MSG003

10708 074072                    ESCAPE  TST                      ; AND ABORT TEST
      074072 104410                    TRAP   C$ESCAPE
      074074 001146                    .WORD  L10056-.

10709
10710 074076                    55$:
10711
10712                    ;2ND PASS PCBB SETUP
10713
10714 074076 012705 012620'          MOV     #WTMOD2,R5              ;DEFAULT MODE FUNCTION
10715 074102 004737 032002'          JSR     PC,LDPBB                ;LOAD FUNCTION -> PCBB
10716 074106 012777 004100 104110    MOV     #DNI!INTE,@PCSR0        ;ENABLE INTERRUPTS
10717 074114 112777 000102 104102    MOVB    #INTE!GETCMD,@PCSR0    ;ISSUE GET_CMD PORT COMMAND
10718 074122 004737 026550'          JSR     PC,CHKDNI                ;DNI?
10719 074126 103010                    BCC     60$                      ;YES, SKIP ERROR REPORT
10720 074130                    FTL

      074130 004737 026652'          JSR     PC,CHKFTL                ; 'FATL' BIT SET?

10721 074134                    ERRHRD  574.,ERR010,MSG003        ;NO, REPORT ERROR
      074134 104456                    TRAP   C$ERHRD
      074136 001076                    .WORD  574
      074140 023331'                    .WORD  ERR010
      074142 021736'                    .WORD  MSG003

10722 074144                    ESCAPE  TST                      ; AND ABORT TEST
      074144 104410                    TRAP   C$ESCAPE
      074146 001074                    .WORD  L10056-.

10723
10724 074150                    ; 60$:
10725 074150 004737 030264'          JSR     PC,CLRDNI                ;WRITE ONE TO CLEAR DNI
10726
10727 074154 103010                    BCC     70$                      ; ERROR?
10728 074156                    FTL                                ; NO

      074156 004737 026652'          JSR     PC,CHKFTL                ; 'FATL' BIT SET?

10729 074162                    ERRHRD  575.,ERR006,MSG003        ; YES, REPORT ERROR
      074162 104456                    TRAP   C$ERHRD
      074164 001077                    .WORD  575
      074166 023030'                    .WORD  ERR006
      074170 021736'                    .WORD  MSG003

;0730 074172                    ESCAPE  TST                      ; AND ABORT TEST
      074172 104410                    TRAP   C$ESCAPE
      074174 001046                    .WORD  L10056-.

10731
10732                    ;WRITE RING FORMAT
    
```

```

10733
10734 074176 012705 012540' 70$: MOV #WTRNGS,R5 ; DEFAULT WRITE RING FORMAT FUNCTION
10735 074202 004737 032002 JSR PC,LDPCCB ; LOAD FUNCTION -> PCBB
10736 074206 012705 012704' MOV #RFRMT,R5 ; DEFAULT RING FORMAT
10737 074212 012700 000006 MOV #6,R0 ; FORMAT = SIX WORDS
10738 074216 004737 032260' JSR PC,LDUDBB ; JAD RING FORMAT -> UDBB
10739 074222 012777 004100 103774 MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
10740 074230 112777 000102 103766 MOVB #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
10741 074236 004737 026550' JSR PC,CHKDNI ; DNI ?
10742 074242 103010 BCC 80$ ; YES
10743 074244 FTL

074244 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

10744 074250 ERRHRD 576.,ERR010,MSG003 ; NO, REPORT ERROR
074250 104456 TRAP C$ERHRD
074252 001100 .WORD 576
074254 023331' .WORD ERR010
074256 021736' .WORD MSG003

10745 074260 ESCAPE TST ; AND ABORT TEST
074260 104410 TRAP C$ESCAPE
074262 000760 .WORD L10056-.

10746 ;
10747 074264 004737 030264' 80$: JSR PC,CLR DNI ; WRITE ONE TO CLEAR DNI
10748 BCC 90$ ; ERROR ?
10749 074270 103010 BCC 90$ ; NO
10750 074272 FTL

074272 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

10751 074276 ERRHRD 577.,ERR006,MSG003 ; YES, REPORT ERROR
074276 104456 TRAP C$ERHRD
074300 001101 .WORD 577
074302 023030' .WORD ERR006
074304 021736' .WORD MSG003

10752 074306 ESCAPE TST ; AND ABORT TEST
074306 104410 TRAP C$ESCAPE
074310 000732 .WORD L10056-.

10753 ;
10754 ;WRITE PHYSICAL ADDRESS
10755 ;
10756 074312 90$: MOV #DEFAULT,R5 ; POINT TO DEFAULT PHYS. ADDR
10757 074312 012705 000272' JSR PC,LDPHYA ; SAVE IN DEFAULT TABLE
10758 074316 004737 032050' MOV #WTPHYA,R5 ; DEFAULT WRITE PHYSICAL ADDR FUNC
10759 074322 012705 012500' JSR PC,LDPCCB ; LOAD FUNCTION -> PCBB
10760 074326 004737 032002' MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
10761 074332 012777 004100 103664 MOVB #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
10762 074340 112777 000102 103656 JSR PC,CHKDNI ; DNI ?
10763 074346 004737 026550' BCC 100$ ; YES
10764 074352 103010 FTL

074354 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

10766 074360 ERRHRD 600.,ERR010,MSG003 ; NO, REPORT ERROR
074360 104456 TRAP C$ERHRD
074362 001130 .WORD 600
  
```

```

    074364 023331'
    074366 021736'
10767 074370          ESCAPE TST          ; AND ABORT TEST          .WORD  ERR010
    074370 104410          ;                               .WORD  MSG003
    074372 000650          ;                               TRAP   C$ESCAPE
    ;                               .WORD  L10056-.
10768
10769 074374 004737 030264' 100$: JSR    PC,CLRDN1          ; WRITE ONE TO CLEAR DNI
10770          ;                               ; ERROR ?
10771 074400 103010          BCC    110$          ; NO
10772 074402          FTL
    074402 004737 026652'          JSR    PC,CHKFTL          ; 'FATL' BIT SET?
10773 074406          ERRHRD 601.,ERR006,MSG003 ; YES, REPORT ERROR
    074406 104456          ;                               TRAP   C$ERHRD
    074410 001131          ;                               .WORD  601
    074412 023030'          ;                               .WORD  ERR006
    074414 021736'          ;                               .WORD  MSG003
10774 074416          ESCAPE TST          ; AND ABORT TEST          TRAP   C$ESCAPE
    074416 104410          ;                               .WORD  L10056-.
    074420 000622          ;
10775          ;SET UP RINGS FOR ONE BUFFER LOOPBACK
10776
10777
10778 074422 012705 014510' 110$: MOV    @TDRB1C,R5          ; DEFAULT ONE BUFFER TRANSMIT RING
10779 074426 004737 032164'          JSR    PC,LDTDRB          ; LOAD TDRB
10780 074432 012705 012750'          MOV    @RDRB1A,R5          ; DEFAULT ONE BUFFER RECEIVE RING
10781 074436 004737 032070'          JSR    PC,LDRDRB          ; LOAD RDRB
10782
10783          ;SET UP BUFFERS AND START
10784
10785          ;IF 1ST PASS WILL LOAD FOR 32 BYTE PACKET, NO CRC
10786
10787 074442 005704          TST    R4                ;1ST PASS?
10788 074444 100405          BMI    115$             ;NO, SKIP 1ST PASS SETUP
10789 074446 012702 000016          MOV    @14.,R2           ;PACKET SIZE (WORDS), NO CRC TO BE ADDED
10790 074452 005037 016562'          CLR    DOCRC             ;NO CRC
10791 074456 000410          BR     117$             ;SKIP 2ND PASS SETUP
10792 074460
10793 074460 012705 014530' 115$: MOV    @TDRB1D,R5          ;SETUP BUFFER TRANSMIT RING
10794 074464 004737 032164'          JSR    PC,LDTDRB          ;LOAD TDRB
10795 074470 012702 000016          MOV    @14.,R2           ;PACKET SIZE (WORDS) CRC TO BE ADDED
10796 074474 005037 016562'          CLR    DOCRC             ;TRANSMITTER TO ADD CRC
10797 074500
10798 074500 004737 032544' 117$: JSR    PC,SETBF           ;LOAD HEADER AND DATA
10799 074504 012777 004100 103512          MOV    @DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
10800 074512 112777 000104 103504          MOVB  @INTE!START,@PCSR0 ; ISSUE START PORT COMMAND
10801 074520 004737 026550'          JSR    PC,CHKDNI          ; DNI?
10802 074524 103010          BCC    120$             ; YES
10803 074526          FTL
    074526 004737 026652'          JSR    PC,CHKFTL          ; 'FATL' BIT SET?
10804 074532          ERRHRD 602.,ERR012,MSG003 ; NO, REPORT ERROR
    074532 104456          ;                               TRAP   C$ERHRD
    074534 001132          ;                               .WORD  602
    074536 023447'          ;                               .WORD  ERR012
  
```





10868	075000				ERRHRD 610.,ERR044		;NO, REPORT ERROR		
	075000	104456						TRAP	C#ERHRD
	075002	001142						.WORD	610
	075004	026175'						.WORD	ERR044
	075006	000000						.WORD	0
10869	075010				ESCAPE TST		; AND EXIT TEST		
	075010	104410						TRAP	C#ESCAPE
	075012	000230						.WORD	L10056--
10870	075014			167\$:					
10871	075014	012705	000620'		MOV #TDRB,R5		;CHECK TDRB		
10872	075020	004737	027500'		JSR PC,CHKTDR		;ERRORS?		
10873	075024	103010			BCC 170\$		;NO		
10874	075026				FTL				
	075026	004737	026652'		JSR PC,CHKFTL		; 'FATL' BIT SET?		
10875	075032				ERRHRD 611.,ERR020,MSG005		;YES, REPORT ERROR		
	075032	104456						TRAP	C#ERHRD
	075034	001143						.WORD	611
	075036	024206'						.WORD	ERR020
	075040	022042'						.WORD	MSG005
10876	075042				ESCAPE TST		; AND ABORT TEST		
	075042	104410						TRAP	C#ESCAPE
	075044	000176						.WORD	L10056--
10877									
10878	075046			170\$:					
10879	075046	012777	004100	103150	MOV #DNI!INTE,@PCSRO		; ENABLE INTERRUPTS		
10880	075054	112777	000117	103142	MOV #INTE!STOP,@PCSRO		; ISSUE STOP PORT COMMAND		
10881	075062	004737	026550'		JSR PC,CHKDNI		; DNI ?		
10882	075066	103010			BCC 180\$		; YES		
10883	075070				FTL				
	075070	004737	026652'		JSR PC,CHKFTL		; 'FATL' BIT SET?		
10884	075074				ERRHRD 612.,ERR019,MSG003		; NO, REPORT ERROR		
	075074	104456						TRAP	C#ERHRD
	075076	001144						.WORD	612
	075100	024126'						.WORD	ERR019
	075102	021736'						.WORD	MSG003
10885	075104				ESCAPE TST		; AND ABORT TEST		
	075104	104410						TRAP	C#ESCAPE
	075106	000134						.WORD	L10056--
10886									
10887	075110	004737	030264'	180\$:	JSR PC,CLR DNI		; WRITE ONE TO CLEAR DNI		
10888							; ERROR ?		
10889	075114	103010			BCC 190\$		; NO		
10890	075116				FTL				
	075116	004737	026652'		JSR PC,CHKFTL		; 'FATL' BIT SET?		
10891	075122				ERRHRD 613.,ERR006,MSG003		; YES, REPORT ERROR		
	075122	104456						TRAP	C#ERHRD
	075124	001145						.WORD	613
	075126	023030'						.WORD	ERR006
	075130	021736'						.WORD	MSG003
10892	075132				ESCAPE TST		; AND ABORT TEST		
	075132	104410						TRAP	C#ESCAPE



```
075134 000106 .WORD L10056-.
10893 075136 190$:
10894
10895 ;IF DONE ONLY 1 PASS MUST GO FOR 2ND
10896
10897 075136 004737 030236' JSR PC,CLRCV ;CLEAR RECEIVE BUFFER
10898 075142 005704 TST R4 ;2ND PASS?
10899 075144 100402 BMI 200$ ;YES, EXIT TEST
10900 075146 000137 073716' JMP 35$ ;NO, DO 2ND PASS
10901 075152 200$:
10902 075152 004737 030346' JSR PC,CLINTR ;INSURE DELUA INTR BITS CLEAR
10903
10904 075156 EXIT TST
075156 104432 TRAP C$EXIT
075160 000062 .WORD L10056-.
10905
10906 ;LOCAL TEST MESSAGE
10907
10908 075162 104 105 114 T24ID: .ASCIZ 'DELUA INTERNAL LOOPBACK TRANSMIT LENGTH ERROR '
075165 125 101 040
075170 111 116 124
075173 105 122 116
075176 101 114 040
075201 114 117 117
075204 120 102 101
075207 103 113 040
075212 124 122 101
075215 116 123 115
075220 111 124 040
075223 114 105 116
075226 107 124 110
075231 040 105 122
075234 122 117 122
075237 040 000
10909 .EVEN
10910
10911 075242 ENDTST
075242 L10056: TRAP C$ETST
075242 104401
10912
10913
10914
```

10916  
 10917  
 10918  
 10919  
 10920  
 10921  
 10922  
 10923  
 10924  
 10925  
 10926  
 10927  
 10928  
 10929  
 10930  
 10931  
 10932  
 10933  
 10934  
 10935  
 10936

.SBTTL TEST 25: SIMULTANEOUS OPERATIONS TEST

```

;*****
;
; THIS TEST VERIFIES THAT SIMULTANEOUS OPERATIONS CAN BE PERFORMED.
; AN INTERNAL LOOPBACK WILL BE PERFORMED SIMULTANEOUSLY WITH A READ
; COUNTERS PORT FUNCTION.
;
; TEST SEQUENCE:
; 1. WRITE MODE REGISTER = PPOM and INTERNAL LOOPBACK MODE
; 2. WRITE RING FORMAT
; 3. WRITE PHYSICAL ADDRESS
; 4. SET UP RINGS AND BUFFERS
; 5. SET UP READ COUNTERS FUNCTION
; 6. ISSUE START
; 7. ISSUE GET COMMAND PORT COMMAND
; 8. CHECK FOR ERRORS
; 9. ISSUE STOP
;*****
    
```

10937 075244  
 075244  
 10938  
 10939 075244

```

BGNTST
;
; T25:
;
PNTMAC T25ID
MOV #T25ID,R4 ;GET POINTER TO TEST NAME MESSAGE
JSR PC,PNTID ;PRINT TEST NUMBER AND NAME
    
```

075244 012704 077072'  
 075250 004737 032734'

; END OF MACRO EXPANSION OF 'PNTMAC'

10940 075254 004737 033434'  
 10941 075260 103034  
 10942 075262 012777 004100 102734  
 10943 075270 112777 000140 102726  
 10944 075276 004737 027736'  
 10945 075302 103010  
 10946 075304

```

JSR PC,TINIT ; IS A DEVICE RESET NEEDED?
BCC 30$ ; NO
MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
MOVB #INTE!RSET,@PCSR0 ; YES, RESET DELUA
JSR PC,CKDNI ; DNI ?
BCC 20$ ; YES
FTL
    
```

075304 004737 026652'

```

JSR PC,CHKFTL ; 'FATL' BIT SET?
    
```

10947 075310  
 075310 104456  
 075312 001146  
 075314 023030'  
 075316 021736'

```

ERRHRD 614.,ERR006,MSG003 ; NO, REPORT ERROR
    
```

```

TRAP C$ERHRD
.WORD 614
.WORD ERR006
.WORD MSG003
    
```

10948 075320  
 075320 104410  
 075322 001610

```

ESCAPE TST ; AND ABORT TEST
    
```

```

TRAP C$ESCAPE
.WORD L10057-.
    
```

10949  
 10950 075324 004737 030264'  
 10951  
 10952 075330 103010  
 10953 075332

```

; 20$: JSR PC,CLRDN1 ; WRITE ONE TO CLEAR DNI
; ERROR ?
BCC 30$ ; NO
FTL
    
```

075332 004737 026652'

```

JSR PC,CHKFTL ; 'FATL' BIT SET?
    
```

```

10954 075336          ERRHRD  615.,ERR006,MSG003      ; YES, REPORT ERROR
      075336 104456          TRAP          C$ERHRD
      075340 001147          .WORD          615
      075342 023030'        .WORD          ERR006
      075344 021736'        .WORD          MSG003
10955 075346          ESCAPE  TST                ; AND ABORT TEST
      075346 104410          TRAP          C$ESCAPE
      075350 001562          .WORD          L10057 .
10956
10957 075352          ;
      30$:
10958 075352 004737 030212' JSR      PC,CLRBUF          ; CLEAR TBUF AND RBUF
10959 075356 004737 031732' JSR      PC,LDDFLT         ; LOAD DEFAULT PHY. ADDRESS TABLES
10960 075362 004737 032032' JSR      PC,LDPCSR        ; ADDRESS OF PCBB -> PCSR2!3
10961 075366 012777 004100 102630 MOV     #DNI!INTE,@PCSR0  ; ENABLE INTERRUPTS
10962 075374 112777 000101 102622 MOVB    #INTE!GETPCB,@PCSR0 ; ISSUE GET_PCBB PORT COMMAND
10963 075402 004737 026550' JSR      PC,CHKDNI        ; DNI?
10964 075406 103010        BCC     40$              ; YES
10965 075410          FTL
      075410 004737 026652' JSR      PC,CHKFTL          ; 'FATL' BIT SET?
10966 075414          ERRHRD  616.,ERR009,MSG003      ; NO, REPORT ERROR
      075414 104456          TRAP          C$ERHRD
      075416 001150          .WORD          616
      075420 023245'        .WORD          ERR009
      075422 021736'        .WORD          MSG003
10967 075424          ESCAPE  TST                ; AND ABORT TEST
      075424 104410          TRAP          C$ESCAPE
      075426 001504          .WORD          L10057-.
10968
10969 075430 004737 030264' ;
      40$: JSR      PC,CLRDN1          ; WRITE ONE TO CLEAR DNI
10970          BCC     50$              ; ERROR ?
10971 075434 103010        BCC     50$              ; NO
10972 075436          FTL
      075436 004737 026652' JSR      PC,CHKFTL          ; 'FATL' BIT SET?
10973 075442          ERRHRD  617.,ERR006,MSG003      ; YES, REPORT ERROR
      075442 104456          TRAP          C$ERHRD
      075444 001151          .WORD          617
      075446 023030'        .WORD          ERR006
      075450 021736'        .WORD          MSG003
10974 075452          ESCAPE  TST                ; AND ABORT TEST
      075452 104410          TRAP          C$ESCAPE
      075454 001456          .WORD          L10057 .
10975
10976          ;
10977          ;WRITE MODE REGISTER = INTERNAL LOOPBACK AND PROM MODE
10978 075456 012705 012600' 50$: MOV     #WTMODE,R5          ; DEFAULT WRITE MODE FUNCTION
10979 075462 004737 032002' JSR      PC,LDPBB         ; LOAD FUNCTION -> PCBB
10980 075466 012777 004100 102530 MOV     #DNI!INTE,@PCSR0  ; ENABLE INTERRUPTS
10981 075474 112777 000102 102522 MOVB    #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
10982 075502 004737 026550' JSR      PC,CHKDNI        ; DNI ?
10983 075506 103010        BCC     60$              ; YES
10984 075510          FTL
      075510 004737 026652' JSR      PC,CHKFTL          ; 'FATL' BIT SET?

```

```

10985 075514          ERRHRD  620.,ERR010,MSG003      ; NO, REPORT ERROR
      075514 104456
      075516 001154
      075520 023331'
      075522 021736'
10986 075524          ESCAPE  TST                ; AND ABORT TEST
      075524 104410
      075526 001404
10987
10988 075530 004737 030264'      ; 60$: JSR      PC,CLRDN1      ; WRITE ONE TO CLEAR DNI
10989
10990 075534 103010          BCC      70$                ; ERROR ?
10991 075536          FTL
      075536 004737 026652'      JSR      PC,CHKFTL          ; NO
      075536 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
10992 075542          ERRHRD  621.,ERR006,MSG003      ; YES, REPORT ERROR
      075542 104456
      075544 001155
      075546 023030'
      075550 021736'
10993 075552          ESCAPE  TST                ; AND ABORT TEST
      075552 104410
      075554 001356
10994
10995          ;WRITE RING FORMAT
10996 075556 012705 012540'      70$:  MOV      @WTRNGS,R5      ; DEFAULT WRITE RING FORMAT FUNCTION
10997 075562 004737 032002'      JSR      PC,LDPCCB          ; LOAD FUNCTION -> PCBB
10998 075566 012705 012734'      MOV      @RFRMTE,R5        ; DEFAULT RING FORMAT
10999 075572 012700 000006      MOV      @6,RO              ; FORMAT = SIX WORDS
11000 075576 004737 032260'      JSR      PC,LDJDBB          ; LOAD RING FORMAT -> UDBB
11001 075602 012777 004100 102414  MOV      @DNI!INTE,@PCSR0    ; ENABLE INTERRUPTS
11002 075610 112777 000102 102406  MOV      @INTE!GETCMD,@PCSR0 ; ISSUE GET CMD PORT COMMAND
11003 075616 004737 026550'      JSR      PC,CHKDNI          ; DNI ?
11004 075622 103010          BCC      80$                ; YES
11005 075624          FTL
      075624 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
11006 075630          ERRHRD  622.,ERR010,MSG003      ; NO, REPORT ERROR
      075630 104456
      075632 001156
      075634 023331'
      075636 021736'
11007 075640          ESCAPE  TST                ; AND ABORT TEST
      075640 104410
      075642 001270
11008
11009 075644 004737 030264'      ; 80$: JSR      PC,CLRDN1      ; WRITE ONE TO CLEAR DNI
11010
11011 075650 103010          BCC      90$                ; ERROR ?
11012 075652          FTL
      075652 004737 026652'      JSR      PC,CHKFTL          ; NO
      075652 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
11013 075656          ERRHRD  623.,ERR006,MSG003      ; YES, REPORT ERROR
    
```

```

075656 104456
075660 001157
075662 023030'
075664 021736'
11014 075666          ESCAPE TST          ; AND ABORT TEST
075666 104410
075670 001242
11015
11016          ;WRITE PHYSICAL ADDRESS
11017
11018 075672
11019 075672 012705 000272'
11020 075676 004737 032050'
11021 075702 012705 012500'
11022 075706 004737 032002'
11023 075712 012777 004100 102304
11024 075720 112777 000102 102276
11025 075726 004737 026550'
11026 075732 103010
11027 075734          FTL

          075734 004737 026652'
          JSR      PC,CHKFTL          ; 'FATL' BIT SET?
11028 075740          ERRHRD 624.,ERR010,MSG003 ; NO, REPORT ERROR
075740 104456
075742 001160
075744 023331'
075746 021736'
11029 075750          ESCAPE TST          ; AND ABORT TEST
075750 104410
075752 001160
11030
11031 075754 004737 030264'
11032          ;100$: JSR      PC,CLRDNI          ; WRITE ONE TO CLEAR DNI
11033 075760 103010          BCC      110$          ; ERROR ?
11034 075762          FTL          ; NO

          075762 004737 026652'
          JSR      PC,CHKFTL          ; 'FATL' BIT SET?
11035 075766          ERRHRD 625.,ERR006,MSG003 ; YES, REPORT ERROR
075766 104456
075770 001161
075772 023030'
075774 021736'
11036 075776          ESCAPE TST          ; AND ABORT TEST
075776 104410
076000 001132
11037
11038          ;SET UP RING BUFFER LOOPBACK
11039
11040 076002 012705 014740'
11041 076006 004737 032222'
11042 076012 012705 013570'
11043 076016 004737 032126'
11044
11045          ;SET UP BUFFERS
11046
          TRAP      C$ERHRD
          .WORD     623
          .WORD     ERR006
          .WORD     MSG003
          TRAP      C$ESCAPE
          .WORD     L10057-.
          TRAP      C$ERHRD
          .WORD     624
          .WORD     ERR010
          .WORD     MSG003
          TRAP      C$ESCAPE
          .WORD     L10057-.
          TRAP      C$ERHRD
          .WORD     625
          .WORD     ERR006
          .WORD     MSG003
          TRAP      C$ESCAPE
          .WORD     L10057.
          ;100$: MOV      #TDRBXX,R5          ; DEFAULT BUFFER TRANSMIT RING
          JSR      PC,LDTDRX          ; LOAD TDRB
          MOV      #RDRBXX,R5          ; DEFAULT BUFFER RECEIVE RING
          JSR      PC,LDRDRX          ; LOAD RDRB
  
```

```

11047 076022 005037 016562' CLR DOCRC ; NO APPEND CRC
11048 076026 012737 000006 016560' MOV #6,BYTCNT ; BYTES/PACKET
11049 076034 004737 033006' JSR PC,SETBUF ; SET UP BUFFERS
11050
11051 ;SET UP READ COUNTER FUNCTION
11052
11053 076040 012705 012550' MOV #RDCNT,R5 ; DEFAULT READ COUNTERS FUNCTION
11054 076044 004737 032002' JSR PC,LDPCBB ; LOAD FUNCTION -> PCBB
11055 076050 005037 000314' CLR UDBB+4 ; INSURE RECEIVED PACKET COUNTER
11056 ; IS CLEAR
11057
11058 ;CLEAR INTERRUPT BITS
11059
11060 076054 012777 175400 102142 MOV #CLINTB,@PCSR0 ; CLEAR INTERRUPT BITS
11061
11062 ;
11063 ;ISSUE START
11064 076062 112777 000004 102134 MOVB #START,@PCSR0 ; ISSUE START PORT COMMAND
11065 076070 004737 026550' JSR PC,CHKDNI ; DNI?
11066 076074 103010 BCC 112# ; YES, SKIP ERROR REPORT
11067 076076 FTL
;
076076 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?
;
11068 076102 ERRHRD 626.,ERR012,MSG003 ; REPORT ERROR
;
076102 104456 TRAP C$ERHRD
076104 001162 .WORD 626
076106 023447' .WORD ERR012
076110 021736' .WORD MSG003
;
11069 076112 ESCAPE TST ; AND ABORT TEST
;
076112 104410 TRAP C$ESCAPE
076114 001016 .WORD L10057-.
;
11070
11071 076116 112# :
11072 076116 004737 030264' JSR PC,CLR DNI ; CLEAR DNI
11073 076122 103010 BCC 113# ; IF DNI CLEARED SKIP ERROR REPORT
11074 076124 FTL
;
076124 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?
;
11075 076130 ERRHRD 627.,ERR006,MSG003 ; ELSE REPORT ERROR
;
076130 104456 TRAP C$ERHRD
076132 001163 .WORD 627
076134 023030' .WORD ERR006
076136 021736' .WORD MSG003
;
11076 076140 ESCAPE TST ; AND ABORT TEST
;
076140 104410 TRAP C$ESCAPE
076142 000770 .WORD L10057 .
;
11077 ;
11078 ;ISSUE GET COMMAND FOR READ COUNTERS FUNCTION
11079
11080 076144 113# :
11081 076144 112777 000002 102052 MOVB #GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
11082
11083 ;WAIT FOR DNI BEFORE CONTINUING
11084
11085 076152 004737 026550' JSR PC,CHKDNI ; CHECK FOR DNI

```

11086	076156	103010		BCC	118\$		; SKIP ERROR REPORT IF DNI		
11087	076160			FTL					
	076160	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
11088	076164			ERRHRD	630.,ERR010,MSG003		; REPORT ERROR	TRAP	C#ERHRD
	076164	104456						WORD	630
	076166	001166						.WORD	ERR010
	076170	023331'						.WORD	MSG003
	076172	021736'							
11089	076174			ESCAPE	TST		; AND ABORT TEST	TRAP	C#ESCAPE
	076174	104410						.WORD	L10057-.
	076176	000734							
11090	076200					118\$:			
11091	076200	004737	030264'	JSR	PC,CLRDNI		; CLEAR DNI		
11092	076204	103010		BCC	125\$		; CONTINUE IF OK		
11093	076206			FTL					
	076206	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
11094	076212			ERRHRD	631.,ERR006,MSG003		; ELSE REPORT ERROR	TRAP	C#ERHRD
	076212	104456						.WORD	631
	076214	001167						.WORD	ERR006
	076216	023030'						.WORD	MSG003
	076220	021736'							
11095	076222			ESCAPE	TST		; AND ABORT TEST	TRAP	C#ESCAPE
	076222	104410						.WORD	L10057-.
	076224	000706							
11096									
11097	076226	152777	000100	101770	125\$:	BISB	#INTE,#PCSR0		; ENABLE INTERRUPTS
11098									
11099	076234	004737	027566'			JSR	PC,CHKTXI		; TXI ?
11100	076240	103010				BCC	140\$		; YES
11101	076242					FTL			
	076242	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
11102	076246			ERRHRD	632.,ERR013,MSG003		; NO, REPORT ERROR	TRAP	C#ERHRD
	076246	104456						.WORD	632
	076250	001170						.WORD	ERR013
	076252	023530'						.WORD	MSG003
	076254	021736'							
11103	076256			ESCAPE	TST		; AND ABORT TEST	TRAP	C#ESCAPE
	076256	104410						.WORD	L10057-.
	076260	000652							
11104									
11105	076262	004737	030500'			140\$:	JSR	PC,CLRTXI	; WRITE ONE TO CLEAR TXI
11106									; ERROR ?
11107	076266	103010				BCC	150\$		; NO
11108	076270					FTL			
	076270	004737	026652'	JSR	PC,CHKFTL		; 'FATL' BIT SET?		
11109	076274			ERRHRD	633.,ERR014,MSG003		; YES, REPORT FRROR	TRAP	C#ERHRD
	076274	104456						.WORD	633
	076276	001171						.WORD	ERR014
	076300	023561'							





```

076434 000476                                .WORD  L10057-.
11138
11139 076436 012705 016264                    164$: MOV    #TDR14A,R5      ; POINT TO EXPECTED TDRB
11140 076442 004737 032370'                  JSR    PC,LDXTDR          ; LOAD INTO XTDRBO TABLE
11141 076446 012705 001314'                  MOV    #TDRX+192.,R5     ; CHECK TDRB
11142 076452 004737 027500'                  JSR    PC,CHK'DR         ; ERRORS ?
11143 076456 103010                            BCC    190$              ; NO
11144 076460                                FTL

076460 004737 026652'                        JSR    PC,CHKFTL         ; 'FATL' BIT SET?

11145 076464                                ERRHRD 637.,ERR034,MSG005 ; YES, REPORT ERROR
076464 104456                                TRAP   C$ERHRD
076466 001175                                .WORD 637
076470 025255'                                .WORD ERR034
076472 022042'                                .WORD MSG005
11146 076474                                ESCAPE TST              ; AND ABORT TEST
076474 104410                                TRAP   C$ESCAPE
076476 000434                                .WORD L10057-.

11147
11148 ;CHECK 1ST RING ENTRY
11149
11150 076500 012705 001624'                    190$: MOV    #RDRX,R5      ; CHECK RDRB OWNERSHIP
11151 076504 004737 027024'                  JSR    PC,CHKOWN        ; OWN = PORT DRIVER ?
11152 076510 103010                            BCC    200$              ; YES
11153 076512                                FTL

076512 004737 026652'                        JSR    PC,CHKFTL         ; 'FATL' BIT SET?

11154 076516                                ERRHRD 640.,ERR030      ; NO, REPORT ERROR
076516 104456                                TRAP   C$ERHRD
076520 001200                                .WORD 640
076522 024753'                                .WORD ERR030
076524 000000                                .WORD 0
11155 076526                                ESCAPE TST              ; AND ABORT TEST
076526 104410                                TRAP   C$ESCAPE
076530 000402                                .WORD L10057 .

11156
11157 076532 012705 016454'                    ;
11158 076536 004737 032340'                    200$: MOV    #RDR20C,R5   ; POINT TO EXPECTED RDRB
11159 076542 012705 001624'                  JSR    PC,LDXRDR        ; LOAD INTO XRDRBO TABLE
11160 076546 004737 027206'                  MOV    #RDRX,R5         ; CHECK RDRB
11161 076552 103010                            JSR    PC,CHKRDR        ; ERRORS ?
11162 076554                                BCC    202$              ; NO
11162 076554                                FTL

076554 004737 026652'                        JSR    PC,CHKFTL         ; 'FATL' BIT SET?

11163 076560                                ERRHRD 641.,ERR036,MSG006 ; YES, REPORT ERROR
076560 104456                                TRAP   C$ERHRD
076562 001201                                .WORD 641
076564 025345'                                .WORD ERR036
076566 022204'                                .WORD MSG006
11164 076570                                ESCAPE TST              ; AND ABORT TEST
076570 104410                                TRAP   C$ESCAPE
076572 000340                                .WORD L10057 .

11165
11166 ;CHECK LAST RING ENTRY
    
```

```

11167
11168 076574 012705 002130'      204$:  MOV    #RDRX+196.,R5      ; CHECK RDRB OWNERSHIP
11169 076600 004737 027024'      JSR    PC,CHKOWN      ; OWN = PORT DRIVER ?
11170 076604 103010                BCC    204$          ; YES
11171 076606                FTL

      076606 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?

11172 076612                ERRHRD  642.,ERR031    ; NO, REPORT ERROR
      076612 104456                TRAP   C$ERHRD
      076614 001202                .WORD  642
      076616 025060'                .WORD  ERR031
      076620 000000                .WORD  0

11173 076622                ESCAPE  TST          ; AND ABORT TEST
      076622 104410                TRAP   C$ESCAPE
      076624 000306                .WORD  L10057-.

11174
11175 076626 012705 016454'      204$:  MOV    #RDR20C,R5      ; POINT TO EXPECTED RDRB
11176 076632 004737 032340'      JSR    PC,LDXRDR      ; LOAD INTO XRDRBO TABLE
11177 076636 012705 002124'      MOV    #RDRX+'92.,R5  ; CHECK RDRB
11178 076642 004737 027206'      JSR    PC,CHKRDR      ; ERRORS ?
11179 076646 103010                BCC    210$          ; NO
11180 076650                FTL

      076650 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?

11181 076654                ERRHRD  643.,ERR037,MSG006 ; YES, REPORT EPROR
      076654 104456                TRAP   C$ERHRD
      076656 001203                .WORD  643
      076660 025433'                .WORD  ERR037
      076662 022204'                .WORD  MSG006

11182 076664                ESCAPE  TST          ; AND ABORT TEST
      076664 104410                TRAP   C$ESCAPE
      076666 000244                .WORD  L10057-.

11183
11184
11185                ;COMPARE RBUF WITH TBUF
11186
11187 076670 013705 016560'      210$:  MOV    BYTCNT,R5      ; COMPARE DATA
11188 076674 004737 030712'      JSR    PC,CMPDAT      ; DATA COMPARE ERROR ?
11189 076700 103006                BCC    220$          ; NO
11190 076702                ERRHRD  644.,ERR022,MSG007 ; YES, REPORT ERROR
      076702 104456                TRAP   C$ERHRD
      076704 001204                .WORD  644
      076706 024350'                .WORD  ERR022
      075710 022346'                .WORD  MSG007

11191 076712                ESCAPE  TST          ; AND ABORT TEST
      076712 104410                TRAP   C$ESCAPE
      076714 000216                .WORD  L10057 .

11192
11193                ;
11194                ; 220$:
11194 076716 012705 006472'      MOV    #RBUF+32,R5    ; OFFSET TO CRC
11195 076722 004737 030642'      JSR    PC,CMPCRC      ; ERRORS ?
11196 076726 103006                BCC    225$          ; NO
11197 076730                ERRHRD  645.,ERR023,MSG008 ; YES, REPORT EPROR
      076730 104456                TRAP   C$ERHRD
      076732 001205                .WORD  645
    
```

076734	024417'						.WORD	ERR023
076736	022400'						.WORD	MSG008
11198 076740			ESCAPE	TST				; AND ABORT TEST
076740	104410						TRAP	C\$ESCAPE
076742	000170						.WORD	L10057--
11199								
11200 076744					i			
11201 076744	005737	000314'			225\$:			
11202 076750	001010		TST	UDBB+4				; HAVE RECEIVED AT LEAST ONE PACKET?
11203 076752			BNE	230\$				; YES, 'KIP ERPROR REPORT
			FTL					
	076752	004737	026652'					
			JSR	PC,CHKFTL				; 'FATL' BIT SET?
11204 076756			ERRHRD	646.,ERR046,MSG010				; NO, REPORT ERROR
076756	104456						TRAP	C\$ERHRD
076760	001206						.WORD	646
076762	026337'						.WORD	ERR046
076764	022604'						.WORD	MSG010
11205 076766			ESCAPE	TST				; AND ABORT TEST
076766	104410						TRAP	C\$ESCAPE
076770	000142						.WORD	L10057 .
11206								
11207 076772					i			
11208 076772	012777	004100	101224		230\$:			
11209 077000	112777	000117	101216	MOV	#DNI!INTE,@PCSR0			; ENABLE INTERRUPTS
11210 077006	004737	026550'		MOV	#INTE!STOP,@PCSR0			; ISSUE STOP PORT COMMAND
11211 077012	103010		JSR	PC,CHKDNI				; DNI ?
11212 077014			BCC	240\$				; YES
			FTL					
	077014	004737	026652'					
			JSR	PC,CHKFTL				; 'FATL' BIT SET?
11213 077020			ERRHRD	647.,ERR019,MSG003				; NO, REPORT ERROR
077020	104456						TRAP	C\$ERHRD
077022	001207						.WORD	647
077024	024126'						.WORD	ERR019
077026	021736'						.WORD	MSG003
11214 077030			ESCAPE	TST				; AND ABORT TEST
077030	104410						TRAP	C\$ESCAPE
077032	000100						.WORD	L10057--
11215								
11216 077034	004737	030264'			i			
11217					240\$:			
11218 077040	103010		JSR	PC,CLRDN1				; WRITE ONE TO CLEAR DNI
11219 077042			BCC	250\$				; ERROR ?
			FTL					; NO
	077042	004737	026652'					
			JSR	PC,CHKFTL				; 'FATL' BIT SET?
11220 077046			ERRHRD	650.,ERR006,MSG003				; YES, REPORT ERROR
077046	104456						TRAP	C\$ERHRD
077050	001212						.WORD	650
077052	023030'						.WORD	ERR006
077054	021736'						.WORD	MSG003
11221 077056			ESCAPE	TST				; AND ABORT TEST
077056	104410						TRAP	C\$ESCAPE
077060	000052						.WORD	L10057--
11222 077062								
11223 077062	004737	030346'			i			
			JSR	PC,CLINTR	250\$:			; INSURE DELUA INTR BITS CLEAR

```
11224
11225 077066          EXIT   TST
      077066 104432
      077070 000042          TRAP   C#EXIT
                                .WGRD  L10057-.
11226
11227          ;LOCAL TEST MESSAGE
11228
11229 077072    104    105    114  T25ID: .ASCIZ 'DELUA SIMULTANEOUS OPERATIONS '
      077075    125    101    040
      077100    123    111    115
      077103    125    114    124
      077106    101    116    105
      077111    117    125    123
      077114    040    117    120
      077117    105    122    101
      077122    124    111    117
      077125    116    123    040
      077130    000
11230          .EVEN
11231
11232          ENDTST
      077132
      077132 104401          L10057:  TRAP   C#ETST
```

11234  
 11235  
 11236  
 11237  
 11238  
 11239  
 11240  
 11241  
 11242  
 11243  
 11244  
 11245  
 11246  
 11247  
 11248  
 11249  
 11250  
 11251  
 11252  
 11253  
 11254  
 11255  
 11256  
 11257

.SBTTL TEST 26: EXTERNAL LOOPBACK TEST (REQUIRES INSTALLED LOOPBACK CONN.)

```

;*****
;
; THIS TEST VERIFIES THAT AN EXTERNAL LOOPBACK OPERATION
; CAN BE PERFORMED SUCCESSFULLY. TEST WILL NOT RUN UNLESS
; EXTERNAL LOOPBACK BIT IN P-TABLE HAS PREVIOUSLY BEEN SET.
; THIS TEST WILL ALSO BE SKIPPED IF IN UNATTENDED MODE.
;
; NOTE: IF AN EXTERNAL LOOPBACK IS NOT INSTALLED FOLLOWING
; OPERATOR PROMPT, THE TEST WILL FAIL.
;
; TEST SEQUENCE:
; 1. WRITE MODE REGISTER = EXTERNAL LOOPBACK, PROM MODE
; 2. WRITE RING FORMAT
; 3. WRITE PHYSICAL ADDRESS
; 4. SET UP RINGS AND BUFFERS
; 5. ISSUE START
; 6. CHECK FOR ERRORS
; 7. ISSUE STOP
;*****
    
```

11258 077134  
 077134

BGNTST

T26::

11259  
 11260  
 11261  
 11262  
 11263  
 11264

;IS EXTERNAL LOOPBACK OPERATION DESIRED?

11262 077134 005737 000222'  
 11263 077140 001006  
 11264 077142

```

TST      EXLOOP      ;SELECTED?
BNE      1$          ;YES, CONTINUE WITH TEST
PNTMAC   SKIP        ;PRINT TEST ID AND REASON FOR SKIP
    
```

077142 012704 020775'  
 077146 004737 032734'

```

MOV      #SKIP,R4    ;GET POINTER TO TEST NAME MESSAGE
JSR      PC,PNTID    ;PRINT TEST NUMBER AND NAME
    
```

; END OF MACRO EXPANSION OF 'PNTMAC'

11265 077152  
 077152 104432  
 077154 001622

```

EXIT     TST          ; AND GET OUT OF TEST
                                TRAP      C$EXIT
                                .WORD    L10060--
    
```

11266  
 11267  
 11268  
 11269  
 11270

;IF UNATTENDED MODE WILL NOT RUN THIS TEST

11269 077156  
 11270 077156 104450  
 11271 077160  
 077160 103406  
 11272 077162

```

1$:
    MANUAL
BCOMPLETE      5$          ;IF NOT UAM, CONTINUE TEST
                                TRAP      C$MANI
                                BCS      5$
PNTMAC   SKIP26          ;PRINT TEST ID AND REASON FOR SKIP
    
```

077162 012704 021056'  
 077166 004737 032734'

```

MOV      #SKIP26,R4    ;GET POINTER TO TEST NAME MESSAGE
JSR      PC,PNTID      ;PRINT TEST NUMBER AND NAME
    
```

; END OF MACRO EXPANSION OF 'PNTMAC'

11273 077172

```

EXIT     TST          ; AND GET OUT OF TEST
    
```

```

077172 104432
077174 001602
11274 077176
11275 077176
077176 012704 100744'
077202 004737 032734'
5$: PNTMAC T26ID
MOV #T26ID,R4 ;GET POINTER TO TEST NAME MESSAGE
JSR PC,PNTID ;PRINT TEST NUMBER AND NAME
; END OF MACRO EXPANSION OF 'PNTMAC'
11276
11277
11278
11279 077206
11280 077206
077206 012746 020570'
077212 012746 000001
077216 010600
077220 104417
077222 062706 000004
11281 077226
077226 104443
077230 000406
077232 016622'
077234 000152
077236 020705'
077240 000015
077242 000000
077244 000002
077246
10000$:
;MESSAGE ADDRESS DATA, LOCATION
;(WORD FOR RESPONSE), A FOR ASCII,
;STORE RESPONSE VALUE IN BITS 0,1
;OF REPLY, 0-2 (LIMITS), DEFAULT
;ACCEPTED.
10000$:
;WAIT FOR CORRECT RESPONSE
BCC 10$
11282
11283
11284
11285
11286
11287 077246
077246 103357
BNCOMPLETE 10$
11288
11289 077250 004737 033434'
11290 077254 103034
11291 077256 012777 004100 100740
11292 077264 112777 000140 100732
11293 077272 004737 027736'
11294 077276 103010
11295 077300
JSR PC,TINIT ; IS A DEVICE RESET NEEDED?
BCC 30$ ; NO
MOV #DNI!INTE @PCSR0 ; PRECONDITION INTR EN.
MOVB #INTE!RSET,@PCSR0 ; YES, RESET DELUA
JSR PC,CKDNI ; DNI ?
BCC 20$ ; YES
FTL
077300 004737 026652'
JSR PC,CHKFTL ; 'FATL' BIT SET?
ERRHRD 651.,ERR006,MSG003 ; NO, REPORT ERROR
TRAP C$ERHPD
.WORD 651
.WORD ERR006
.WORD MSG003
11296 077304
077304 104456
077306 001213
077310 023030'
077312 021736'
ESCAPE TST ; AND ABORT TEST
TRAP C$ESCAPE
.WORD L10060
11297 077314
077314 104410
077316 001460
11298

```

```

11299 077320 004737 030264'      20$: JSR    PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
11300                               ; ERROR ?
11301 077324 103010              BCC    30$              ; NO
11302 077326                               FTL

      077326 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?

11303 077332 ERRHRD 652.,ERR006,MSG003 ; YES, REPORT ERROR
      077332 104456                               TRAP   C$ERHRD
      077334 001214                               .WORD 652
      077336 023030'                               .WORD ERR006
      077340 021736'                               .WORD MSG003

11304 077342 ESCAPE TST          ; AND ABORT TEST
      077342 104410                               TRAP   C$ESCAPE
      077344 001432                               .WORD L10C60-.

11305                               ;
11306 077346                               ; 30$:
11307 077346 004737 030212'      JSR    PC,CLRBUF      ; CLEAR XMIT,RCV BUFFERS
11308 077352 004737 031732'      JSR    PC,LDDFLT      ; LOAD DEFAULT PHY.ADDRESS TABLES
11309 077356 004737 032032'      JSR    PC,LDPCSR      ; ADDRESS OF PCBB -> PCSR2!3
11310 077362 012777 004100 100634 MOV    #DNI!INTE,@PCSRO ; PRECONDITION INTR EN.
11311 077370 112777 000101 100626 MOVB   #INTE!GETPCB,@PCSRO ; ISSUE GET_PCBB PORT COMMAND
11312 077376 004737 026550'      JSR    PC,CHKDNI      ; DNI?
11313 077402 103010              BCC    40$              ; YES
11314 077404                               FTL

      077404 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?

11315 077410 ERRHRD 653.,ERR009,MSG003 ; NO, REPORT ERROR
      077410 104456                               TRAP   C$ERHRD
      077412 001215                               .WORD 653
      077414 023245'                               .WORD ERR009
      077416 021736'                               .WORD MSG003

11316 077420 ESCAPE TST          ; AND ABORT TEST
      077420 104410                               TRAP   C$ESCAPE
      077422 001354                               .WORD L10060 .

11317                               ;
11318 077424 004737 030264'      40$: JSR    PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
11319                               ; ERROR ?
11320 077430 103010              BCC    50$              ; NO
11321 077432                               FTL

      077432 004737 026652'      JSR    PC,CHKFTL      ; 'FATL' BIT SET?

11322 077436 ERRHRD 654.,ERR006,MSG003 ; YES, REPORT ERROR
      077436 104456                               TRAP   C$ERHRD
      077440 001216                               .WORD 654
      077442 023030'                               .WORD ERR006
      077444 021736'                               .WORD MSG003

11323 077446 ESCAPE TST          ; AND ABORT TEST
      077446 104410                               TRAP   C$ESCAPE
      077450 001326                               .WORD L10060 .

11324                               ;
11325                               ; WRITE MODE REGISTER = EXTERNAL LOOPBACK AND PROM MODE
11326                               ;
11327 077452 012705 012630'      50$: MOV    #WTMOD3,R5      ; DEFAULT WRITE MODE FUNCTION
11328 077456 004737 032002'      JSR    PC,LDPCCB      ; LOAD FUNCTION -> PCBB
  
```

```

11329 077462 012777 004100 100534      MOV    #DNI!INTE,@PCSR0      ; PRECONDITION INTR EN.
11330 077470 112777 000102 100526      MOVB  #INTE!GETCMD,@PCSR0   ; ISSUE GET_CMD PORT COMMAND
11331 077476 004737 026550'          JSR   PC,CHKDNI             ; DNI ?
11332 077502 103010                    BCC   60$                   ; YES
11333 077504                    FTL

      077504 004737 026652'          JSR   PC,CHKFTL             ; 'FATL' BIT SET?

11334 077510                    ERRHRD 655.,ERR010,MSG003    ; NO, REPORT ERROR
      077510 104456                    TRAP  C$ERHRD
      077512 001217                    .WORD 655
      077514 023331'                   .WORD ERR010
      077516 021736'                   .WORD MSG003

11335 077520                    ESCAPE TST                   ; AND ABORT TEST
      077520 104410                    TRAP  C$ESCAPE
      077522 001254                    .WORD L10060-.

11336 ;
11337 077524 004737 030264'          ;60$: JSR   PC,CLRDN1         ; WRITE ONE TO CLEAR DNI
11338 ;                                ; ERROR ?
11339 077530 103010                    BCC   70$                   ; NO
11340 077532                    FTL

      077532 004737 026652'          JSR   PC,CHKFTL             ; 'FATL' BIT SET?

11341 077536                    ERRHRD 656.,ERR006,MSG003    ; YES, REPORT ERROR
      077536 104456                    TRAP  C$ERHRD
      077540 001220                    .WORD 656
      077542 023030'                   .WORD ERR006
      077544 021736'                   .WORD MSG003

11342 077546                    ESCAPE TST                   ; AND ABORT TEST
      077546 104410                    TRAP  C$ESCAPE
      077550 001226                    .WORD L10060-.

11343 ;
11344 ;WRITE RING FORMAT
11345 ;
      70$: MOV    #WTRNGS,R5         ; DEFAULT WRITE RING FORMAT FUNCTION
11346 077552 012705 012540'          JSR   PC,LDPCCB             ; LOAD FUNCTION -> PCBB
11347 077556 004737 032002'          MOV   #FRMT,R5              ; DEFAULT RING FORMAT
11348 077562 012705 012704'          MOV   #6,R0                 ; FORMAT = SIX WORDS
11349 077566 012700 000006          JSR   PC,LDUDBB             ; LOAD RING FORMAT -> UDBB
11350 077572 004737 032260'          MOV   #DNI!INTE,@PCSR0     ; PRECONDITION INTR EN.
11351 077576 012777 004100 100420      MOVB  #INTE!GETCMD,@PCSR0   ; ISSUE GET_CMD PORT COMMAND
11352 077604 112777 000102 100412      JSR   PC,CHKDNI             ; DNI ?
11353 077612 004737 026550'          BCC   80$                   ; YES
11354 077616 103010                    FTL

      077620 004737 026652'          JSR   PC,CHKFTL             ; 'FATL' BIT SET?

11356 077624                    ERRHRD 657.,ERR010,MSG003    ; NO, REPORT ERROR
      077624 104456                    TRAP  C$ERHRD
      077626 001221                    .WORD 657
      077630 023331'                   .WORD ERR010
      077632 021736'                   .WORD MSG003

11357 077634                    ESCAPE TST                   ; AND ABORT TEST
      077634 104410                    TRAP  C$ESCAPE
      077636 001140                    .WORD L10060-.

11358 ;
  
```



```

11359 077640 004737 030264'      80$:   JSR      PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
11360                                     ; ERROR ?
11361 077644 103010                BCC      90$           ; NO
11362 077646                                     FTL

      077646 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?

11363 077652 ERRHRD 658.,ERR006,MSG003 ; YES, REPORT ERROR
      077652 104456 TRAP      C$ERHRD
      077654 001222 .WORD    658
      077656 023030' .WORD    ERR006
      077660 021736' .WORD    MSG003

11364 077662 ESCAPE TST           ; AND ABORT TEST
      077662 104410 TRAP      C$ESCAPE
      077664 001112 .WORD    L10060-.

11365                                     ;
11366                                     ;WRITE PHYSICAL ADDRESS
11367
11368 077666      90$:
11369 077666 012705 000272'      MOV      #DFULT,R5      ; GET DEFAULT PHYSICAL ADDRESS
11370 077672 004737 032050'      JSR      PC,LDPHYA      ; PLACE IT IN DATA TABLE
11371 077676 012705 012500'      MOV      #WTPHYA,R5     ; DEFAULT WRITE PHYSICAL ADDR FUNC
11372 077702 004737 032002'      JSR      PC,LPCBB       ; LOAD FUNCTION -> PCBB
11373 077706 012777 004100 100310 MOV      #DNI!INTE,@PCSR0 ; PRECONDITION INTR EN.
11374 077714 112777 000102 100302 MOVB     #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
11375 077722 004737 026550'      JSR      PC,CHKDNI      ; DNI ?
11376 077726 103010                BCC      100$          ; YES
11377 077730                                     FTL

      077730 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?

11378 077734 ERRHRD 660.,ERR010,MSG003 ; NO, REPORT ERROR
      077734 104456 TRAP      C$ERHRD
      077736 001224 .WORD    660
      077740 023331' .WORD    ERR010
      077742 021735' .WORD    MSG003

11379 077744 ESCAPE TST           ; AND ABORT TEST
      077744 104410 TRAP      C$ESCAPE
      077746 001030 .WORD    L10060-.

11380                                     ;
11381 077750 004737 030264'      100$:  JSR      PC,CLRDNI      ; WRITE ONE TO CLEAR DNI
11382                                     ; ERROR ?
11383 077754 103010                BCC      110$          ; NO
11384 077756                                     FTL

      077756 004737 026652'      JSR      PC,CHKFTL      ; 'FATL' BIT SET?

11385 077762 ERRHRD 661.,ERR006,MSG003 ; YES, REPORT ERROR
      077762 104456 TRAP      C$ERHRD
      077764 001225 .WORD    661
      077766 023030' .WORD    ERR006
      077770 021736' .WORD    MSG003

11386 077772 ESCAPE TST           ; AND ABORT TEST
      077772 104410 TRAP      C$ESCAPE
      077774 001002 .WORD    L10060 .

11387                                     ;
11388                                     ;SET UP RINGS FOR ONE BUFFER LOOPBACK
    
```

```

11389
11390 077776 012705 014410'      110$: MOV    #TDRB1A,R5      ; DEFAULT ONE BUFFER TRANSMIT RING
11391 100002 004737 032164'      JSR    PC,LDTDRB          ; LOAD TORB
11392 100006 012705 012750'      MOV    #RDRB1A,R5      ; DEFAULT ONE BUFFER RECEIVE RING
11393 100012 004737 032070'      JSR    PC,LDRDRB          ; LOAD RDRB
11394
11395      ;SET UP BUFFERS AND START
11396
11397 100016 005037 016562'      CLR    D0CRC            ; NO APPEND CRC
11398 100022 012737 000006 016560'  MOV    #6,BYTCNT        ; DATA BYTE COUNT
11399 100030 004737 033006'      JSR    PC,SETBUF        ; SET UP BUFFERS
11400
11401      ;INSURE SOURCE AND DESTINATION ADDRESSES = DEFAULT PHYSICAL ADDRESS
11402
11403 100034 012737 000004 000300'  MOV    #4,PCBB          ; READ DEFAULT PHYSICAL ADDRESS
11404 100042 012777 004100 100154  MOV    #DNI!INTE,@PCSR0 ; PRECONDITION INTR ENABLE
11405 100050 112777 000102 100146  MOVB   #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
11406 100056 004737 026550'      JSR    PC,CHKDNI        ; DNI?
11407 100062 103010                BCC    112$            ; YES
11408 100064                FTL
11409 100064 004737 026652'      JSR    PC,CHKFTL        ; 'FATL' BIT SET?
11409 100070                ERRHRD 662.,ERR012,MSG003
11409 100070 104456                TRAP  C$ERRHRD
11409 100072 001226                .WORD 662
11409 100074 023447'                .WORD ERR012
11409 100076 021736'                .WORD MSG003
11410 100100                ESCAPE TST
11410 100100 104410                TRAP  C$ESCAPE
11410 100102 000674                .WORD L10060 .
11411 100104                112$:
11412 100104 004737 030264'      JSR    PC,CLRDN1        ; CLEAR DNI
11413 100110 103010                BCC    114$
11414 100112 004737 026652'      JSR    PC,CHKFTL        ; CHECK FOR FATL BIT SET
11415 100116                ERRHRD 663.,ERR006,MSG003
11415 100116 104456                TRAP  C$ERRHRD
11415 100120 001227                .WORD 663
11415 100122 023030'                .WORD ERR006
11415 100124 021736'                .WORD MSG003
11416 100126                ESCAPE TST
11416 100126 104410                ; EXIT TEST
11416 100130 000646                TRAP  C$ESCAPE
11417 100132                .WORD L10060-.
11418
11419      ;LOAD DEFAULT PHYSICAL ADDRESS INTO SOURCE AND DESTINATION ADDRESS
11420
11421 100132 012700 000002                MOV    #2,R0            ; INIT COUNTER
11422 100136 012701 002436'      MOV    #TBUF,R1        ; BASE ADDRESS OF XMIT BUFFER
11423 100142                116$:
11424 100142 013721 000302'      MOV    PCBB+2,(R1)+    ; READ OUT
11425 100146 013721 000304'      MOV    PCBB+4,(R1)+    ; PHYSICAL
11426 100152 013721 000306'      MOV    PCBB+6,(R1)+    ; ADDRESS
11427 100156 077007                SOB    R0,116$        ; DO TWICE
11428
11429      ;SEND PACKET
11430
    
```

11431	100160	012777	004100	100036	MOV	@DNI!INTE,@PCSRO	; PRECONDITION INTR EN.		
11432	100166	112777	000104	100030	MOVB	@INTE!START,@PCSRO	; ISSUE START PORT COMMAND		
11433	100174	004737	026550'		JSR	PC,CHKDNI	; DNI?		
11434	100200	103010			BCC	120\$	; YES		
11435	100202				FTL				
	100202	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
11436	100206				ERRHRD	664.,ERR012,MSG003	; NO, REPORT ERROR	TRAP	C\$ERHRD
	100206	104456						.WORD	664
	100210	001230						.WORD	ERR012
	100212	023447'						.WORD	MSG003
	100214	021736'							
11437	100216				ESCAPE	TST	; AND ABORT TEST	TRAP	C\$ESCAPE
	100216	104410						.WORD	L10060
	100220	000556							
11438									
11439	100222	004737	030264'		JSR	PC,CLRDN1	; WRITE ONE TO CLEAR DNI		
11440							; ERROR ?		
11441	100226	103010			BCC	130\$	; NO		
11442	100230				FTL				
	100230	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
11443	100234				ERRHRD	665.,ERR006,MSG003	; YES, REPORT ERROR	TRAP	C\$ERHRD
	100234	104456						.WORD	665
	100236	001231						.WORD	ERR006
	100240	023030'						.WORD	MSG003
	100242	021736'							
11444	100244				ESCAPE	TST	; AND ABORT TEST	TRAP	C\$ESCAPE
	100244	104410						.WORD	L10060-
	100246	000530							
11445									
11446	100250	004737	027566'		JSR	PC,CHKTXI	; TXI ?		
11447	100254	103010			BCC	140\$	; YES		
11448	100256				FTL				
	100256	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
11449	100262				ERRHRD	666.,ERR013,MSG003	; NO, REPORT ERROR	TRAP	C\$ERHRD
	100262	104456						.WORD	666
	100264	001232						.WORD	ERR013
	100266	023530'						.WORD	MSG003
	100270	021736'							
11450	100272				ESCAPE	TST	; AND ABORT TEST	TRAP	C\$ESCAPE
	100272	104410						.WORD	L10050-
	100274	000502							
11451									
11452	100276	004737	030500'		JSR	PC,CLR1XI	; WRITE ONE TO CLEAR TXI		
11453							; ERROR ?		
11454	100302	103010			BCC	150\$	; NO		
11455	100304				FTL				
	100304	004737	026652'		JSR	PC,CHKFTL	; 'FATL' BIT SET?		
11456	100310				ERRHRD	667.,ERR014,MSG003	; YES, REPORT ERROR	TRAP	C\$ERHRD
	100310	104456							

	100312	001233					.WORD	667
	100314	023561'					.WORD	ERR014
	100316	021736'					.WORD	MSG003
11457	100320		ESCAPE	TST				; AND ABORT TEST
	100320	104410					TRAP	C\$ESCAPE
	100322	000454					.WORD	L10060
11458								
11459	100324	012705	000620'	i 150\$:	MOV	#TDRB,R5		; CHECK TDRB OWNERSHIP
11460	100330	004737	027024'		JSR	PC,CHKOWN		; OWN = PORT DRIVER ?
11461	100334	03010			BCC	160\$		; YES
11462	100336				FTL			
	100336	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
11463	100342		ERRHRD	668.,ERR018				; NO, REPORT ERROR
	100342	104456					TRAP	C\$ERHRD
	100344	001234					.WORD	668
	100346	024026'					.WORD	ERR018
	100350	000000					.WORD	0
11464	100352		ESCAPE	TST				; AND ABORT TEST
	100352	104410					TRAP	C\$ESCAPE
	100354	000422					.WORD	L10060-
11465								
11466	100356	012705	016264'	i 160\$:	MOV	#TDR14A,R5		; POINT TO EXPECTED TDRB
11467	100362	004737	032370'		JSR	PC,LDXTDR		; LOAD INTO XTDRBO TABLF
11468	100366	012705	000620'		MOV	#TDRB,R5		; CHECK TDRB
11469	100372	004737	027500'		JSR	PC,CHKTDR		; ERRORS ?
11470	100376	103010			BCC	170\$		; NO
11471	100400				FTL			
	100400	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
11472	100404		ERRHRD	670.,ERR020,MSG005				; YES, REPORT ERROR
	100404	104456					TRAP	C\$ERHRD
	100406	001236					.WORD	670
	100410	024206'					.WORD	ERR020
	100412	022042'					.WORD	MSG005
11473	100414		ESCAPE	TST				; AND ABORT TEST
	100414	104410					TRAP	C\$ESCAPE
	100416	000360					.WORD	L10060-
11474								
11475	100420	004737	027316'	i 170\$:	JSR	PC,CHKRXI		; RXI ?
11476	100424	103010			BCC	180\$		; YES
11477	100426				FTL			
	100426	004737	026652'		JSR	PC,CHKFTL		; 'FATL' BIT SET?
11478	100432		ERRHRD	671.,ERR015,MSG003				; NO, REPORT ERROR
	100432	104456					TRAP	C\$ERHRD
	100434	001237					.WORD	671
	100436	023627'					.WORD	ERR015
	100440	021736'					.WORD	MSG003
11479	100442		ESCAPE	TST				; AND ABORT TEST
	100442	104410					TRAP	C\$ESCAPE
	100444	000332					.WORD	L10060-
11480								
11481	100446	004737	030432'	i 180\$:	JSR	PC,CLRRXI		; WRITE ONE TO CLEAR RXI

```

11482                                     ; ERROR ?
11483 100452 103010                      BCC 190$ ; NO
11484 100454                               FTL

      100454 004737 026652'              JSR PC,CHKFTL ; 'FATL' BIT SET?

11485 100460                               ERRHRD 672.,ERR016,MSG003 ; YES, REPORT ERROR
      100460 104456                               TRAP C$ERHRD
      100462 001240                               .WORD 672
      100464 023660'                               .WORD ERR016
      100466 021736'                               .WORD MSG003

11486 100470                               ESCAPE TST ; AND ABORT TEST
      100470 104410                               TRAP C$ESCAPE
      100472 000304                               .WORD L10060-.

11487                                     ;
11488 100474 012705 000660'              190$: MOV #RDRB,R5 ; CHECK RDRB OWNERSHIP
11489 100500 004737 027024'              JSR PC,CHKOWN ; OWN = PORT DRIVER ?
11490 100504 103010                      BCC 200$ ; YES
11491 100506                               FTL

      100506 004737 026652'              JSR PC,CHKFTL ; 'FATL' BIT SET?

11492 100512                               ERRHRD 673.,ERR017 ; NO, REPORT ERROR
      100512 104456                               TRAP C$ERHRD
      100514 001241                               .WORD 673
      100516 023726'                               .WORD ERR017
      100520 000000                               .WORD 0

11493 100522                               ESCAPE TST ; AND ABORT TEST
      100522 104410                               TRAP C$ESCAPE
      100524 000252                               .WORD L10060-.

11494                                     ;
11495 100526 012705 016454'              200$: MOV #RDR20C,R5 ; POINT TO EXPECTED RDRB
11496 100532 004737 032340'              JSR PC,LDXRDR ; LOAD INTO XRDRBO TABLE
11497 100536 012705 000660'              MOV #RDRB,R5 ; CHECK RDRB
11498 100542 004737 027206'              JSR PC,CHKRDR ; ERRORS ?
11499 100546 103010                      BCC 210$ ; NO
11500 100550                               FTL

      100550 004737 026652'              JSR PC,CHKFTL ; 'FATL' BIT SET?

11501 100554                               ERRHRD 674.,ERR021,MSG006 ; YES, REPORT ERROR
      100554 104456                               TRAP C$ERHRD
      100556 001242                               .WORD 674
      100560 024267'                               .WORD ERR021
      100562 022204'                               .WORD MSG006

11502 100564                               ESCAPE TST ; AND ABORT TEST
      100564 104410                               TRAP C$ESCAPE
      100566 000210                               .WORD L10060 .

11503                                     ;
11504                                     ;COMPARE RBUF WITH TBUF
11505                                     ;
11506 100570 013705 016560'              210$: MOV BYTCNT,R5 ; NUMBER OF DATA COMPARES
11507 100574 004737 030712'              JSR PC,CMPDAT ; DATA COMPARE ERROR ?
11508 100600 103006                      BCC 220$ ; NO
11509 100602                               ERRHRD 675.,ERR022,MSG007 ; YES, REPORT ERROR
      100602 104456                               TRAP C$ERHRD
      100604 001243                               .WORD 675
    
```

	100606	024350'							.WORD	ERR022
	100610	022346'							.WORD	MSG007
11510	100612		ESCAPE	TST						
	100612	104410							TRAP	C\$ESCAPE
	100614	000162							.WORD	L10060 .
11511										
11512	100616				220\$:					
11513	100616	012705	006472'	MOV	#RBUF+32,R5					
11514										
11515	100622	004737	030642'	JSR	PC,CMPCRC					
11516	100626	103006		BCC	230\$					
11517	100630			ERRHRD	676.,ERR023,MSG008					
	100630	104456							TRAP	C\$ERHRD
	100632	001244							.WORD	676
	100634	024417'							.WORD	ERR023
	100636	022400'							.WORD	MSG008
11518	100640		ESCAPE	TST						
	100640	104410							TRAP	C\$ESCAPE
	100642	000134							.WORD	L10060-.
11519										
11520	100644	012777	004100	077352	230\$:	MOV	#DNI!INTE,@PCSR0			
11521	100652	112777	000117	077344		MOVB	#INTE!STOP,@PCSR0			
11522	100660	004737	026550'			JSR	PC,CHKDNI			
11523	100664	103010				BCC	240\$			
11524	100666					FTL				
	100666	004737	026652'			JSR	PC,CHKFTL			
11525	100672					ERRHRD	677.,ERR019,MSG003			
	100672	104456								
	100674	001245							TRAP	C\$ERHRD
	100676	024126'							.WORD	677
	100700	021736'							.WORD	ERR019
11526	100702		ESCAPE	TST						
	100702	104410							TRAP	C\$ESCAPE
	100704	000072							.WORD	L10060-.
11527										
11528	100706	004737	030264'	240\$:		JSR	PC,CLRDN1			
11529										
11530	100712	103010				BCC	250\$			
11531	100714					FTL				
	100714	004737	026652'			JSR	PC,CHKFTL			
11532	100720					ERRHRD	680.,ERR006,MSG003			
	100720	104456								
	100722	001250							TRAP	C\$ERHRD
	100724	023030'							.WORD	680
	100726	021736'							.WORD	ERR006
11533	100730		ESCAPE	TST						
	100730	104410							TRAP	C\$ESCAPE
	100732	000044							.WORD	L10060-.
11534	100734				250\$:					
11535	100734	004737	030346'			JSR	PC,CLINTR			
11536										
11537	100740		EXIT	TST						
	100740	104432							TRAP	C\$EXIT

```
100742 000034 .WORD L10060-.
11538
11539 ;LOCAL TEST MESSAGE
11540
11541 100744 104 105 114 T26ID:.ASCIZ 'DELUA EXTERNAL LOOPBACK '
      100747 125 101 040
      100752 105 130 124
      100755 105 122 116
      100760 101 114 040
      100763 114 117 117
      100766 120 102 101
      100771 103 113 040
      100774 000
11542 .EVEN
11543
11544 100776 ENDTST
      100776
      100776 104401 L10060: TRAP C#ETST
```

11546  
 11547  
 11548  
 11549  
 11550  
 11551  
 11552  
 11553  
 11554  
 11555  
 11556  
 11557  
 11558  
 11559  
 11560  
 11561

.SBTTL TEST 27: 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
*****
  
```

```

11562 101000          BGNTST
      101000
11563 101000 005737 016612' TST FRSTIM          ; RUN THIS TEST ? T27::
11564 101004 001006      BNE 5$          ; YES
11565 101006          PNTMAC T27SKP
      101006 012704 102116' MOV #T27SKP,R4      ;GET POINTER TO TEST NAME MESSAGE
      101012 004737 032734' JSR PC,PNTID        ;PRINT TEST NUMBER AND NAME
:
: END OF MACRO EXPANSION OF 'PNTMAC'
11566 101016          EXIT TST          ; NO, EXIT
      101016 104432
      101020 001142          TRAP C$EXIT
:                                     .WORD L10061-.
11567 5$:
11568 101022          PNTMAC T27ID
11569 101022          MOV #T27ID,R4      ;GET POINTER TO TEST NAME MESSAGE
      101022 012704 102060' JSR PC,PNTID        ;PRINT TEST NUMBER AND NAME
      101026 004737 032734'
:
: END OF MACRO EXPANSION OF 'PNTMAC'
11570 101032 004737 033434' JSR PC,TINIT      ; IS A DEVICE RESET NEEDED?
11571 101036 103034      BCC 30$          ; NO
11572 101040 012777 004100 077156 MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
11573 101046 112777 000140 077150 MOVB #INTE!RSET,@PCSR0 ; YES, RESET DELUA
11574 101054 004737 027736' JSR PC,CKDNI      ; DNI ?
11575 101060 103010      BCC 20$          ; YES
11576 101062          FTL
      101062 004737 026652' JSR PC,CHKFTL      ; 'FATL' BIT SET?
11577 101066          ERRHRD 681.,ERR006,MSG003 ; NO, REPORT ERROR
      101066 104456          TRAP C$ERHRD
      101070 001251          .WORD 681
      101072 023030'        .WORD ERR006
      101074 021736'        .WORD MSG003
11578 101076          ESCAPE TST          ; AND ABORT TEST
      101076 104410          TRAP C$ESCAPE
      101100 001062          .WORD L10061 .
  
```



```

11579
11580 101102 004737 030264' 20$: JSR PC,CLRDN1 ; WRITE ONE TO CLEAR DNI
11581 ; ERROR ?
11582 101106 103010 BCC 30$ ; NO
11583 101110 FTL

101110 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

11584 101114 ERRHRD 682.,ERR006,MSG003 ; YES, REPORT ERROR
101114 104456 TRAP C$ERHRD
101116 001252 .WORD 682
101120 023030' .WORD ERR006
101122 021736' .WORD MSG003
11585 101124 ESCAPE TST ; AND ABORT TEST
101124 104410 TRAP C$ESCAPE
101126 001034 .WORD L10061-.

11586
11587 101130 004737 032032' 30$: JSR PC,LDPCSR ; ADDRESS OF PCBB > PCSR2'3
11588 101134 012777 004100 077062 MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
11589 101142 112777 000101 077054 MOVB #INTE!GETPCB,@PCSR0 ; ISSUE GET_PCBB PORT COMMAND
11590 101150 004737 026550' JSR PC,CHKDNI ; DNI?
11591 101154 103010 B>C 40$ ; YES
11592 101156 FTL

101156 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

11593 101162 ERRHRD 683.,ERR009,MSG003 ; NO, REPORT ERROR
101162 104456 TRAP C$ERHRD
101164 001253 .WORD 683
101166 023245' .WORD ERR009
101170 021736' .WORD MSG003
11594 101172 ESCAPE TST ; AND ABORT TEST
101172 104410 TRAP C$ESCAPE
101174 000766 .WORD L10061-.

11595
11596 101176 004737 030264' 40$: JSR PC,CLRDN1 ; WRITE ONE TO CLEAR DNI
11597 ; ERROR ?
11598 101202 103010 BCC 50$ ; NO
11599 101204 FTL

101204 004737 026652' JSR PC,CHKFTL ; 'FATL' BIT SET?

11600 101210 ERRHRD 684.,ERR006,MSG003 ; YES, REPORT ERROR
101210 104456 TRAP C$ERHRD
101212 001254 .WORD 684
101214 023030' .WORD ERR006
101216 021736' .WORD MSG003
11601 101220 ESCAPE TST ; AND ABORT TEST
101220 104410 TRAP C$ESCAPE
101222 000740 .WORD L10061-.

11602
11603 ;READ DEFAULT PHYSICAL ADDRESS
11604
11605 101224 012705 012460' 50$: MOV #RDDEFA,R5 ; READ DEFAULT PHYA FUNCTION
11606 101230 004737 032002' JSR PC,LDPCBB ; LOAD FUNCTION -> PCBB
11607 101234 012777 004100 076762 MOV #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
11608 101242 112777 000102 076754 MOVB #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
  
```

```

11609 101250 004737 026550'      JSR    PC,CHKDNI      ; DNI ?
11610 101254 103010              BCC    60$           ; YES
11611 101256              FTL

      101256 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?
11612 101262              ERRHRD 685.,ERR010,MSG003 ; NO, REPORT ERROR
      101262 104456              TRAP  C$ERHRD
      101264 001255              .WORD 685
      101266 023331'           .WORD ERR010
      101270 021736'           .WORD MSG003
11613 101272              ESCAPE TST           ; AND ABORT TEST
      101272 104410              TRAP  C$ESCAPE
      101274 000666              .WORD L10061-.
11614
11615 101276 004737 030264'      ;
11616              60$:      JSR    PC,CLRDN1     ; WRITE ONE TO CLEAR DNI
11617 101302 103010              BCC    70$           ; ERROR ?
11618 101304              FTL                 ; NO
      101304 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?
11619 101310              ERRHRD 686.,ERR006,MSG003 ; YES, REPORT ERROR
      101310 104456              TRAP  C$ERHRD
      101312 001256              .WORD 686
      101314 023030'           .WORD ERR006
      101316 021736'           .WORD MSG003
11620 101320              ESCAPE TST           ; AND ABORT TEST
      101320 104410              TRAP  C$ESCAPE
      101322 000640              .WORD L10061-.
11621
11622              ;MOVE DEFAULT PHYSICAL ADDRESS FROM PCBB -> DPA
11623
11624 101324 013737 000302' 102164' 70$:      MOV    PCBB+2,DPA
11625 101332 013737 000304' 102166'      MOV    PCBB+4,DPA+2
11626 101340 013737 000306' 102170'      MOV    PCBB+6,DPA+4
11627
11628              ;LOAD ASCII MESSAGE (DEFADR)
11629
11630 101346 004737 031236'      JSR    PC,HEXDP      ; CONVERT TO ASCII HEX
11631
11632              ;READ MICROCODE REVISION
11633
11634 101352 012705 012644'      100$:      MOV    #RDSTA,R5      ; READ PORT STATUS FUNCTION
11635 101356 004737 032002'      JSR    PC,L0PCBB     ; LOAD FUNCTION -> PCBB
11636 101362 012777 004100 076634      MOV    #DNI!INTE,@PCSR0 ; ENABLE INTERRUPTS
11637 101370 112777 000102 076626      MOVB   #INTE!GETCMD,@PCSR0 ; ISSUE GET_CMD PORT COMMAND
11638 101376 004737 026550'      JSR    PC,CHKDNI     ; DNI ?
11639 101402 103010              BCC    110$          ; YES
11640 101404              FTL
      101404 004737 026652'      JSR    PC,CHKFTL     ; 'FATL' BIT SET?
11641 101410              ERRHRD 687.,ERR010,MSG003 ; NO, REPORT ERROR
      101410 104456              TRAP  C$ERHRD
      101412 001257              .WORD 687
      101414 023331'           .WORD ERR010
    
```

```

11642 101416 021736'
101420          ESCAPE TST          ; AND ABORT TEST          .WORD  MSG003
101420 104410          TRAP          C$ESCAPE
101422 000540          .WORD          L10061-.

11643
11644 101424 004737 030264'      ;
11645          ;
11646 101430 103010          BCC 120$          ; WRITE ONE TO CLEAR DNI
11647 101432          FTL          ; ERROR ?
          ; NO

          101432 004737 026652'      JSR PC,CHKFTL          ; 'FATL' BIT SET?

11648 101436          ERRHRD 690.,ERR006,MSG003          ; YES, REPORT ERROR
101436 104456          TRAP          C$ERHRD
101440 001262          .WORD          690
101442 023030'          .WORD          ERR006
101444 021736'          .WORD          MSG003

11649 101446          ESCAPE TST          ; AND ABORT TEST          TRAP  C$ESCAPE
101446 104410          .WORD          L10061 .
101450 000512

11650          ;
11651          ;MOVE MICROCODE REVISION FROM PCBB -> RREV
11652
11653 101452 013737 000302' 102172' 120$:  MOV  PCBB+2,RREV
11654 101460 042737 177700 102172'      BIC  #177700,RREV          ; MASK RREV
11655          ;
11656          ;READ SWITCH PACK
11657
11658 101466 012705 012664'      130$:  MOV  #DMPMEM,R5          ; DEFAULT DUMP INTERNAL MEMORY
11659 101472 004737 032002'      JSR  PC,LDPCBB          ; LOAD FUNCTION -> PCBB
11660 101476 012705 016500'      MOV  #UDB28A,R5          ; DEFAULT UDBB
11661 101502 012700 000005          MOV  #5,R0          ; FIVE WORDS
11662 101506 004737 032260'      JSR  PC,LDUDBB          ; LOAD INTO UDBB
11663 101512 012737 102174' 000312'  MOV  #SWPACK,UDBB+2          ; LOAD ADDRESS
11664 101520 012737 000002 000316'  MOV  #2,UDBB+6          ; LOAD INTERNAL
11665 101526 012737 000030 000320'  MOV  #30,UDBB+10          ; ADDRESS (6000002 OCTAL)
11666 101534 012777 004100 076462  MOV  #DNI!INTE,@PCSR0          ; ENABLE INTERRUPTS
11667 101542 112777 000102 076454  MOVB #INTE!GETCMD,@PCSR0          ; ISSUE GET COMMAND PORT COMMAND
11668 101550 004737 026550'      JSR  PC,CHKDNI          ; DNI ?
11669 101554 103010          BCC 140$          ; YES
11670 101556          FTL

          101556 004737 026652'      JSR  PC,CHKFTL          ; 'FATL' BIT SET?

11671 101562          ERRHRD 691.,ERR010,MSG003          ; NO, REPORT ERROR
101562 104456          TRAP          C$ERHRD
101564 001263          .WORD          691
101566 023331'          .WORD          ERR010
101570 021736'          .WORD          MSG003

11672 101572          ESCAPE TST          ; AND ABORT TEST          TRAP  C$ESCAPE
101572 104410          .WORD          L10061 .
101574 000366

11673
11674 101576 004737 030264'      ;
11675          ;
11676 101602 103010          BCC 150$          ; WRITE ONE TO CLEAR DNI
11677 101604          FTL          ; ERROR ?
          ; NO
  
```

```

101604 004737 026652'      JSR      PC,CHKFTL          ; 'FATL' BIT SET?
11678 101610                ERRHRD  692.,ERR006,MSG003  ; YES, REPORT ERROR
101610 104456                TRAP      C$ERHRD
101612 001264                .WORD   692
101614 023030'              .WORD   ERR006
101616 021736'              .WORD   MSG003
11679 101620                ESCAPE  TST                ; AND ABORT TEST
101620 104410                TRAP      C$ESCAPE
101622 000340                .WORD   L10061-.

11680
11681                ;GET SWITCH PACK INFO READY TO PRINT
11682
11683 101624 013704 102174' 150$:  MOV      SWPACK,R4          ; SWITCH PACK -> R4
11684 101630 005704                TST      R4                ; MFG MODE ENABLED?
11685 101632 100404                BMI      160$              ; NO
11686 101634 012737 102437' 102176'  MOV      @LPMSG1,LPMMSG    ; POINT TO MFG MODE EN MSG
11687 101642 000403                BR       170$              ; SKIP DISABLED MSG
11688 101644 012737 102362' 102176' 160$:  MOV      @LPMSG0,LPMMSG    ; POINT TO MFG MODE DISABLED MSG
11689 101652 013704 102174' 170$:  MOV      SWPACK,R4          ; SWITCH PACK -> R4
11690 101656 042704 117777                BIC      @117777,R4        ; MASK BITS 14 AND 13
11691 101662 012700 000014                MOV      @12.,R0          ; SHIFT BITS FOR INDEX
11692 101666 006204                180$:  ASR      R4
11693 101670 005300                DEC      R0
11694 101672 001375                BNE      180$
11695 101674 062704 102320'                ADD      @BTTBL,R4        ; INDEX INTO BOOT TABLE
11696 101700 011437 102200'                MOV      (R4),BTMSG       ; LOAD INTO BOOT MESSAGE
11697
11698                ; PRINT
11699                ;
11700 101704                PRINTB  @FRM015,@DEFHDR    ; PRINT DEFAULT PHYSICAL ADDRESS
101704 012746 102204'                MOV      @DEFHDR,-(SP)
101710 012746 020177'                MOV      @FRM015,-(SP)
101714 012746 000002                MOV      @2,-(SP)
101720 010600                MOV      SP,R0
101722 104414                TRAP      C$PNTB
101724 062706 000006                ADD      @6,SP
11701 101730                PRINTB  @FRM016,RREV      ; PRINT MICROCODE REV
101730 013746 102172'                MOV      RREV,-(SP)
101734 012746 020204'                MOV      @FRM016,-(SP)
101740 012746 000002                MOV      @2,-(SP)
101744 010600                MOV      SP,R0
101746 104414                TRAP      C$PNTB
101750 062706 000006                ADD      @6,SP
11702 101754                PRINTB  @FRM015,@SWHDR    ; PRINT SWITCH PACK HEADER
101754 012746 102330'                MOV      @SWHDR,-(SP)
101760 012746 020177'                MOV      @FRM015,-(SP)
101764 012746 000002                MOV      @2,-(SP)
101770 010600                MOV      SP,R0
101772 104414                TRAP      C$PNTB
101774 062706 000006                ADD      @6,SP
11703 102000                PRINTB  @FRM015,LPMMSG    ; PRINT LOOPBACK MESSAGE
102000 013746 102176'                MOV      LPMMSG,-(SP)
102004 012746 020177'                MOV      @FRM015,-(SP)
102010 012746 000002                MOV      @2,-(SP)
102014 010600                MOV      SP,R0
    
```

102016	104414								
11704 102020	062706	000006						TRAP	C:PNTB
11704 102024			PRINTB	#FRM015,9TMSG			; PRINT BOOT MESSAGE	ADD	#6,SP
102024	013746	102200'						MOV	BMSG,-(SP)
102030	012746	020177'						MOV	#FRM015,-(SP)
102034	012746	000002						MOV	#2,-(SP)
102040	010600							MOV	SP,RO
102042	104414							TRAP	C:PNTB
102044	062706	000006						ADD	#6,SP
11705									
11706 102050									
11707 102050	004737	030346'							
11708									
11709 102054			JSR	PC,CLINTR			; INSURE DELUA INTR BITS CLEAR		
102054	104432		EXIT	TST					
102056	000104							TRAP	C:EXIT
								.WORD	L10061-
11710									
11711							;LOCAL TEST MESSAGES		
11712									
11713 102060	104	105	114	T27ID:	.ASCIZ 'DELUA PRINT DEVICE PARAMETER '				
102063	125	101	040						
102066	120	122	111						
102071	116	124	040						
102074	104	105	126						
102077	111	103	105						
102102	040	120	101						
102105	122	101	115						
102110	105	124	105						
102113	122	040	000						
11714									
11715 102116	124	110	111	.EVEN	T27SKP:.ASCIZ 'THIS TEST PERFORMED 1ST PASS ONLY '				
102121	123	040	124						
102124	105	123	124						
102127	040	120	105						
102132	122	106	117						
102135	122	115	105						
102140	104	040	061						
102143	123	124	040						
102146	120	101	123						
102151	123	040	117						
102154	116	114	131						
102157	040	000							
11716				.EVEN					
11717 102162				ENDTST					
102162									
102162	104401							L10061:	TRAP C:ETST

```

11719 ;LOCAL STORAGE FOR TEST 27
11720 102164 000000 DPA: .WORD 0 ; DEFAULT PHYSICAL ADDRESS (15:00)
11721 102166 000000 .WORD 0 ; DEFAULT PHYSICAL ADDRESS (31:16)
11722 102170 000000 .WORD 0 ; DEFAULT PHYSICAL ADDRESS (47:32)
11723 ;
11724 102172 000000 RREV: .WORD 0 ; MICROCODE REVISION
11725 ;
11726 102174 000000 SWPACK: .WORD 0 ; SWITCH PACK CONTENTS
11727 102176 000000 LPMSG: .WORD 0 ; LOOPBACK MESSAGE ADDRESS
11728 102200 000000 BTMSG: .WORD 0 ; BOOT MESSAGE ADDRESS
11729 ;
11730 102202 000 HEXDAT: .BYTE 0 ; HEX DATA FOR CONVERSION
11731 102203 000 HEXVAL: .BYTE 0 ; ASCII HEX VALUE
11732 ;
11733 102204 015 012 105 DEFHDR: .ASCII <15><12>/ETHERNET DEFAULT ADDRESS (HEX): /
      102207 124 110 105
      102212 122 116 105
      102215 124 040 104
      102220 105 106 101
      102223 125 114 124
      102226 040 101 104
      102231 104 122 105
      102234 123 123 040
      102237 050 110 105
      102242 130 051 072
      102245 040 040
11734 102247 040 040 DEFADR: .ASCII / /
11735 102251 055 .ASCII /-/
11736 102252 040 040 .ASCII / /
11737 102254 055 .ASCII /-/
11738 102255 040 040 .ASCII / /
11739 102257 055 .ASCII /-/
11740 102260 040 040 .ASCII / /
11741 102262 055 .ASCII /-/
11742 102263 040 040 .ASCII / /
11743 102265 055 .ASCII /-/
11744 102266 040 040 .ASCII / /
11745 102270 015 012 000 .ASCIZ <15><12>
11746 ;
11747 102273 060 NEXTHBL: .ASCII /0/
11748 102274 061 .ASCII /1/
11749 102275 062 .ASCII /2/
11750 102276 063 .ASCII /3/
11751 102277 064 .ASCII /4/
11752 102300 065 .ASCII /5/
11753 102301 066 .ASCII /6/
11754 102302 067 .ASCII /7/
11755 102303 070 .ASCII /8/
11756 102304 071 .ASCII /9/
11757 102305 101 .ASCII /A/
11758 102306 102 .ASCII /B/
11759 102307 103 .ASCII /C/
11760 102310 104 .ASCII /D/
11761 102311 105 .ASCII /E/
11762 102312 106 .ASCII /F/
11763 .EVEN
11764 ;
    
```

```

11765 ;LOOP MESSAGE TABLE
11766 102314 102362' LPTBL: .WORD LPMSG0
11767 102316 102437' .WORD LPMSG1
11768 ;BOOT MESSAGE TABLE
11769 102320 102513' BTTBL: .WORD BTMSG0
11770 102322 102627' .WORD BTMSG2
11771 102324 102551' .WORD BTMSG1
11772 102326 102513' .WORD BTMSG0
11773 ;ASCII MESSAGES
11774 102330 015 012 123 SWHDR: .ASCII <15><12>/SWITCH PACK SET FOR :/
      102333 127 111 124
      102336 103 110 040
      102341 120 101 103
      102344 113 040 123
      102347 105 124 040
      102352 106 117 122
      102355 040 072
11775 102357 015 012 000
11776 102362 040 040 040 LPMSG0: .ASCII / .ASCIZ <15><12>
      102365 040 040 123 SELF TEST MANUFACTURING MODE DISABLED/
      102370 105 114 106
      102373 040 124 105
      102376 123 124 040
      102401 115 101 116
      102404 125 106 101
      102407 103 124 125
      102412 122 111 116
      102415 107 040 115
      102420 117 104 105
      102423 040 104 111
      102426 123 101 102
      102431 114 105 104
11777 102434 015 012 000
11778 102437 040 040 040 LPMSG1: .ASCII / .ASCIZ <15><12>
      102442 040 040 123 SELF TEST MANUFACTURING MODE ENABLED/
      102445 105 114 106
      102450 040 124 105
      102453 123 124 040
      102456 115 101 116
      102461 125 106 101
      102464 103 124 125
      102467 122 111 116
      102472 107 040 115
      102475 117 104 105
      102500 040 105 116
      102503 101 102 114
      102506 105 104
11779 102510 015 012 000
11780 102513 040 040 040 BTMSG0: .ASCII / .ASCIZ <15><12>
      102516 040 040 116 NO REMOTE BOOT ENABLED/
      102521 117 040 122
      102524 105 115 117
      102527 124 105 040
      102532 102 117 117
      102535 124 040 105
      102540 116 101 102
      102543 114 105 104
    
```

11781	102546	015	012	000	
11782	102551	040	040	040	BTMSG1: .ASCII /
	102554	040	040	122	.ASCIZ <15><12>
	102557	105	115	117	REMOTE BOOT, WITH SYSTFM LOAD, ENABLED/
	102562	124	105	040	
	102565	102	117	117	
	102570	124	054	040	
	102573	127	111	124	
	102576	110	040	123	
	102601	131	123	124	
	102604	105	115	040	
	102607	114	117	101	
	102612	104	054	040	
	102615	105	116	101	
	102620	102	114	105	
	102623	104			
11783	102624	015	012	000	
11784	102627	040	040	040	BTMSG2: .ASCII /
	102632	040	040	122	.ASCIZ <15><12>
	102635	105	115	117	REMOTE BOOT ENABLED WITH ROM/
	102640	124	105	040	
	102643	102	117	117	
	102646	124	040	105	
	102651	116	101	102	
	102654	114	105	104	
	102657	040	127	111	
	102662	124	110	040	
	102665	122	117	115	
11785	102670	015	012	000	.ASCIZ <15><12>
11786					.EVEN



```

11789 .TITLE PARAMETER CODING
11800
11801 .SBTTL HARDWARE PARAMETER CODING SECTION
11820
11821
11822
11823 ;**
11824 ; THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
11825 ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES. THE
11826 ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
11827 ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES. THE
11828 ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
11829 ; WITH THE OPERATOR.
11830 ;--
11831 102674 BGNHRD
11831 102674 000010
11831 102676 .WORD L10062-L$HARD/2
11831 102676 L$HARD::
11832
11833 102676 GPRMA ASKCSR,0,0,174510,174610,NO ;FIRST P-TABLE QUESTION
11833 102676 000021 .WORD T$CODE
11833 102700 102716' .WORD ASKCSR
11833 102702 174510 .WORD T$LOLIM
11833 102704 174610 .WORD T$HILIM
11834
11835
11836 102706 GPRMA ASKVEC,2,0,120,220,NO ;SECOND P-TABLE QUESTION
11836 102706 001021 .WORD T$CODE
11836 102710 102751' .WORD ASKVEC
11836 102712 000120 .WORD T$LOLIM
11836 102714 000220 .WORD T$HILIM
11837
11838 102716 ENDHRD
11838 102716 .EVEN
11838 102716 L10062:
11839
11840 102716 127 110 101 ASKCSR: .ASCIZ /WHAT IS THE PCSRO ADDRESS?/
11840 102721 124 040 111
11840 102724 123 040 124
11840 102727 110 105 040
11840 102732 100 103 123
11840 102735 122 060 040
11840 102740 101 104 104
11840 102743 122 105 123
11840 102746 123 077 000
11841 102751 127 110 101 ASKVEC: .ASCIZ /WHAT IS THE VECTOR ADDRESS?/
11841 102754 124 040 111
11841 102757 123 040 124
11841 102762 110 105 040
11841 102765 126 105 103
11841 102770 124 117 122
11841 102773 040 101 104
11841 102776 104 122 105
11841 103001 123 123 077
11841 103004 000
11842 .EVEN
    
```

```

11844 .SBTTL SOFTWARE PARAMETER CODING SECTION
11845
11846 ;**
11847 ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
11848 ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES. THE
11849 ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
11850 ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES. THE
11851 ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
11852 ; WITH THE OPERATOR.
11853 ;--
11854
11855 103006 BGNSFT
11856 103006 000003 .WORD L10063-L$SOFT/2
11857 103010 L$SOFT::
11858 103010 000130 .WORD T$CODE
11859 103012 103016' .WORD ASKEXT
11860 103014 000001 .WORD 1
11861 .EVEN
11862 103016 ENDSFT
11863 103016 L10063: .EVEN
11864 103016 122 125 116 ASKEXT: .ASCIZ /RUN TEST 26 IN EXTERNAL LOOPBACK MODE ?/
11865 103021 040 124 105
11866 103024 123 124 040
11867 103027 062 066 040
11868 103032 111 116 040
11869 103035 105 130 124
11870 103040 105 122 116
11871 103043 101 114 040
11872 103046 114 117 117
11873 103051 120 102 101
11874 103054 103 113 040
11875 103057 115 117 104
11876 103062 105 040 077
11877 103065 000
11878 .EVEN
11879 $PATCH:
11880 103066 .BLKW 20
11881 103066
11882 103126 LASTAD
11883 103126 000000 .EVEN
11884 103130 000000 .WORD 0
11885 103132 .WORD 0
11886 L$LAST::
11887 103132 ENDMOD
11888 .END
11889 000001
11890

```

ADR	000020 G	CLKBR	000250R	C\$INLP	000020	ERDRB0	016520R	ETDRB2	016542R
ADR21	016060R	CLKCSR	000246R	C\$MANI	000050	ERDRB2	016522R	ETDRB4	016544R
ADR21C	016066R	CLKFRE	000254R	C\$MAP	000102	ERDRB4	016524R	ETDRB6	016546R
ASKCSR	102716R	CLKSRV	034430RG	C\$MEM	000031	ERDRB6	016526R	EVL	000004 G
ASKEXT	103016R	CLKTAB	000246R	C\$MMU	000103	ERRBLK	016656RG	EXLOOP	000222R
ASKVEC	102751R	CLKVEC	000252R	C\$MESS	000023	ERRMSG	016654RG	E\$END	002100
ASSEMB	000010	CLRBUF	030212R	C\$OPNR	000034	ERRNBR	016652RG	E\$LOAD	000035
BIT0	000001 G	CLRCNT	012560R	C\$OPNW	000104	ERRS	040000 G	FATL	001000 G
BIT00	000001 G	CLRCV	030236R	C\$PNTB	000014	ERRTYP	016650RG	FATLIB	000002 G
BIT01	000002 G	CLRDNI	030264R	C\$PNTF	000017	ERR001	022660R	FRM001	016752R
BIT02	000004 G	CLRRCE	030364R	C\$PNTS	000016	ERR002	022710R	FRM002	017007R
BIT03	000010 G	CLRRXI	030432R	C\$PNTX	000015	ERR003	022746R	FRM003	017070R
BIT04	000020 G	CLRSER	030546R	C\$PUTB	000072	ERR005	023004R	FRM004	017131R
BIT05	000040 G	CLRSTA	012654R	C\$PUTW	000073	ERR006	023030R	FRM005	017171R
BIT06	000100 G	CLRTXI	030500R	C\$QID	000377	ERR007	023076R	FRM006	017256R
BIT07	000200 G	CLRXTM	030614R	C\$RDBU	000007	ERR008	023166R	FRM007	017343R
BIT08	000400 G	CMODE1	175015 G	C\$REFG	000047	ERR009	023245R	FRM008	017430R
BIT09	001000 G	CMPCRC	030642R	C\$REL	000077	ERR010	023331R	FRM009	017515R
BIT1	000002 G	CMPCRC	030642R	C\$RESE	000033	ERR011	023414R	FRM010	017602R
BIT10	002000 G	CMPCRC	030712R	C\$REVI	000004	ERR012	023447R	FRM011	017667R
BIT11	004000 G	CMPCRC	030772R	C\$REVI	000004	ERR013	023530R	FRM012	017754R
BIT12	010000 G	CMPCRC	031052R	C\$RFLA	000021	ERR014	023530R	FRM013	020041P
BIT13	020000 G	CRCH	015620R	C\$RPT	000025	ERR015	023561R	FRM014	020120R
BIT14	040000 G	C\$AU	000052	C\$SEFG	000046	ERR016	023627R	FRM015	020177R
BIT15	100000 G	C\$AUTO	000061	C\$SPRI	000041	ERR017	023660R	FRM016	020204R
BIT2	000004 G	C\$BRK	000022	C\$SVEC	000037	ERR018	023726R	FRM017	020255R
BIT3	000010 G	C\$BSEG	000004	C\$TOME	000076	ERR019	024026R	FRM018	020303R
BIT4	000020 G	C\$BSUB	000002	DEFADR	102247R	ERR020	024126R	FRM019	020337R
BIT5	000040 G	C\$CLCK	000062	DEFHDR	102204R	ERR021	024206R	FRM020	020373R
BIT6	000100 G	C\$CLEA	000012	DELAY	031132R	ERR022	024267R	FRM021	020427R
BIT7	000200 G	C\$CLOS	000035	DELUAT	000020 G	ERR023	024350R	FRM022	020463R
BIT8	000400 G	C\$CLP1	000006	DEST	000256R	ERR024	024417R	FRM023	020547R
BIT9	001000 G	C\$CPBF	000074	DEVUNI	021310R	ERR025	024444R	FRSTIM	016612R
BLKCR	026506R	C\$CPME	000075	DFAULT	000272R	ERR026	024506R	FTLSET	026726R
BOE	000400 G	C\$CVEC	000036	DFPTBL	000214RG	ERR027	024540R	F\$AU	000015
BTMSG	102200R	C\$DCLN	000044	DIAGMC	000000	ERR028	024540R	F\$AUTO	000020
BTMSG0	102513R	C\$DODU	000051	DMPMEM	012664R	ERR029	024645R	F\$BGN	000040
BTMSG1	102551R	C\$DRPT	000024	DNI	004000 G	ERR030	024753R	F\$CLEA	000007
BTMSG2	102627R	C\$DU	000053	DNIB	000010 G	ERR031	024753R	F\$DU	000016
BTTBL	102320R	C\$EDIT	000000	DNICLR	021575R	ERR032	025060R	F\$END	000041
BUFL	100000 G	C\$ERDF	000055	DNIFLG	016610R	ERR033	025166R	F\$HARD	000004
BYTCNT	016560R	C\$ERHR	000056	DOCRC	016562R	ERR034	025166R	F\$HW	000013
CHKDNI	026550R	C\$ERRO	000060	DPA	102164R	ERR035	025255R	F\$INIT	000006
CHKFTL	026652R	C\$ERSF	000054	DTYPE	002540 G	ERR036	025345R	F\$JMP	000050
CHKOWN	027024R	C\$ERSO	000057	EAFIAG	016606R	ERR037	025345R	F\$MOD	000000
CHKRCE	027102R	C\$ESCA	000010	ECODE	016600R	ERR038	025433R	F\$MSG	000011
CHKRDR	027206R	C\$ESEG	000005	ECRC	016570R	ERR039	025522R	F\$PROT	000021
CHKRXI	027316R	C\$ESUB	000003	ECRCB	016572R	ERR040	025575R	F\$PWR	000017
CHKSER	027676R	C\$ETST	000001	EDAT	016564R	ERR041	025676R	F\$RPT	000012
CHKSTR	027420R	C\$EXIT	000032	EF.CON	000036 G	ERR042	025743R	F\$SEG	000003
CHKTDR	027500R	C\$FREQ	000101	EF.NEW	000035 G	ERR043	026011R	F\$SOFT	000005
CHKTXI	027566R	C\$FRME	000100	EF.PWR	000034 G	ERR044	026076R	F\$SRV	000010
CKDNI	027736R	C\$GETB	000026	EF.RES	000037 G	ERR045	026175R	F\$SUB	000002
CMINTR	030150R	C\$GETW	000027	EF.STA	000040 G	ERR046	026274R	F\$SW	000014
CLBYTE	030224R	C\$GMAN	000043	END13A	016060R	ERR047	026337R	F\$TEST	000001
CLINTB	175400 G	C\$GPHR	000042	ENP	000400 G	ERR048	026415R	GETCMD	000002 G
CLINTR	030346R	C\$GPRI	000040	EPCSR0	016514R	ETDRB0	016540R	GETCRC	031160R
		C\$INIT	000011	EPCSR1	016516R				

GETPCB = 000001 G	I\$SFT = 000041	L\$HPCP 000016RG	L10040 041166R	O\$AU = 000001
GOODST = 000000 G	I\$SRV = 000041	L\$HPTP 000022RG	L10041 041716R	O\$BGNR = 000001
G\$CNTD = 000200	I\$SUB = 000041	L\$HW 000214RG	L10042 043156R	O\$BGNS = 000001
G\$DELM = 000372	I\$TST = 000041	L\$ICP 000104RG	L10043 044572R	O\$DU = 000001
G\$DISP = 000003	J\$JMP = 000167	L\$INIT 033530RG	L10044 046154R	O\$ERRT = 000001
G\$EXCP = 000400	LDBUF 031404R	L\$LADP 000026RG	L10045 047574R	O\$GNSW = 000001
G\$HILI = 000002	LDBUFC 031446R	L\$LAST 103132RG	L10046 050702R	O\$POIN = 000001
G\$LOLI = 000001	LDBUFR 031520R	L\$LOAD 000100RG	L10047 052320R	O\$SETU = 000000
G\$NO = 000000	LDDEST 031704R	L\$LUN 000074RG	L10050 053612R	PATRN1 016626R
G\$OFFS = 000400	LDDFLT 031732R	L\$MREV 000050RG	L10051 055422R	PCBB 000300R
G\$OSI = 000376	LDMEM 012674R	L\$NAME 000000RG	L10052 061104R	PCEI = 040000 G
G\$PRMA = 000001	LDPCBB 032002R	L\$PRIO 000042RG	L10053 065040R	PCEIB = 000100 G
G\$PRMD = 000002	LDPCSR 032032R	L\$PROT 033522RG	L10054 071300R	PCSP0 000224R
G\$PRML = 000000	LDPHYA 032050R	L\$PRT 000112RG	L10055 073576R	PCSR0C 000236R
G\$RADA = 000140	LDRDRB 032070R	L\$REPP 000062RG	L10056 075242R	PCSR0U 000234R
G\$RADB = 000000	LDRDRX 032126R	L\$REV 000010RG	L10057 077132R	PCSR1 000226R
G\$RADD = 000040	LDTDRB 032164R	L\$RPT 033514RG	L10060 100776R	PCSR2 000230R
G\$RADL = 000120	LDTDRX 032222R	L\$SOFT 103010RG	L10061 102162R	PCSR3 000232R
G\$RADO = 000020	LDUDBB 032260R	L\$SPC 000056RG	L10062 102716R	PCT0 = 000200 G
G\$XFER = 000004	LDXCRC 032314R	L\$SPCP 000020RG	L10063 103016R	PDMD = 000010 G
G\$YES = 000010	LDXRDR 032340R	L\$SPTP 000024RG	MEM10A 015634R	PNOP = 000006 G
HALT = 000016 G	LDXTRD 032370R	L\$STA 000030RG	MEM11A 015706R	PNT = 001000 G
HELP = 000000	LOE = 040000 G	L\$SW 000222RG	MEM13A 015712R	PNTID 032734R
HEXDAT 102202R	LOT = 000010 G	L\$TEST 000114RG	METER 016602R	POLYH 016614R
HEXDPA 031236R	LPMSG 102176R	L\$TIML 000014RG	MNMSG1 020705R	POLYHI = 120001 G
HEXH 031320R	LPMSG0 102362R	L\$UNIT 000012RG	MODE15 016464R	POLYL 016616R
HEXL 031356R	LPMSG1 102437R	L10000 000220R	MODE17 016466R	PRI = 002000 G
HEXTBL 102273R	LPTBL 102314R	L10001 000224R	MODE20 016470R	PRIID = 000001 G
HEXVAL 102203R	LSMA 012450R	L10002 021706R	MODE21 016472R	PRI00 = 000000 G
HOE = 100000 G	L\$ACP 000110RG	L10003 021734R	MODE24 016474R	PRI01 = 000040 G
IBE = 010000 G	L\$APT 000036RG	L10004 021766R	MODE25 016476R	PRI02 = 000100 G
IDU = 000040 G	L\$AU 034356RG	L10005 022040R	MSG001 021664RG	PRI03 = 000140 G
IE = 000100 G	L\$AUT 000070RG	L10006 022202R	MSG002 021710RG	PRI04 = 000200 G
IER = 020000 G	L\$AUTO 034340RG	L10007 022344R	MSG003 021736RG	PRI05 = 000240 G
INITH = 000000 G	L\$CCP 000106RG	L10010 022376R	MSG004 021770RG	PRI06 = 000300 G
INTE = 000100 G	L\$CLEA 034342RG	L10011 022460R	MSG005 022042RG	PRI07 = 000340 G
INTMG1 030060R	L\$CO 000032RG	L10012 022602R	MSG006 022204RG	PRNTIT 016620R
INTR = 000200 G	L\$DEPO 000011RG	L10013 022630R	MSG007 022346RG	RBUF 006440R
INTVEC 000240R	L\$DESC 016666RG	L10014 022656R	MSG008 022400RG	RBUF2 007040R
ISR = 000100 G	L\$DESP 000076RG	L10015 033520R	MSG009 022462RG	RBUF3 007440R
ISRDN1 034374RG	L\$DEVP 000060RG	L10017 034336R	MSG010 022604RG	RBUF4 010040R
ISRNXM 034364RG	L\$DISP 000124RG	L10020 034340R	MSG011 022632RG	RBUF5 010440R
IXE = 004000 G	L\$DLY 000116RG	L10021 034346R	MSG1 020570R	RBUF6 011040R
I\$AU = 000041	L\$DTP 000040RG	L10022 034354R	MULTL 016074R	RBUF7 011440R
I\$AUTO = 000041	L\$DTYP 000034RG	L10023 034362R	MULTLC 016170R	RBUF8 012040R
I\$CLN = 000041	L\$DU 034350RG	L10024 034372R	M68FLD 021202R	RCBI = 002000 G
I\$DU = 000041	L\$DUT 000072RG	L10025 034426R	NEXMEM 016604R	RCBIB = 000004 G
I\$HRD = 000041	L\$DVTY 016660RG	L10026 034442R	NIHLT = 000006 G	RDCNT 012550R
I\$INIT = 000041	L\$EF 000052RG	L10027 034622R	NIUNI = 000007 G	RDDEFA 012460R
I\$MOD = 000041	L\$ENVI 000044RG	L10030 035010R	NIUNIB 021406R	RDMODE 012570R
I\$MSG = 000041	L\$ERRT 016650RG	L10031 035216R	NOCLK 021140R	RDMULA 012510R
I\$PROT = 000040	L\$ETP 000102RG	L10032 035404R	NOPF 012440R	RDPHYA 012470R
I\$PTAB = 000041	L\$EXP1 000046RG	L10033 035572R	NORXI 032422R	RDRB 000660R
I\$PWR = 000041	L\$EXP4 000064RG	L10034 036010R	ONEFIL = 000001	RDRBE 000720R
I\$RPT = 000041	L\$EXP5 000066RG	L10035 036226R	ONES = 177777 G	RDRBXX 013570R
I\$SEG = 000041	L\$HARD 102676RG	L10036 036554R	OWN = 100000 G	RDRB1A 012750R
I\$SETU = 000041	L\$HIME 000120RG	L10037 040614R	O\$APTS = 000000	RDRB1B 013010R



. ABS. 000000 000 (RW,I,GBL,ABS,OVR)  
103132 001 (RW,I,LCL,REL,CON)

Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 320  
Work file writes: 325  
Size of work file: 36704 Words ( 144 Pages)  
Size of core pool: 19402 Words ( 74 Pages)  
Operating system: RSX-11M/PLUS (Under VAX/VMS)

Elapsed time: 00:24:28.58  
CZUADA,CZUADA/-SP/CRF/NL:TOC=SVC40/ML,CZUADA