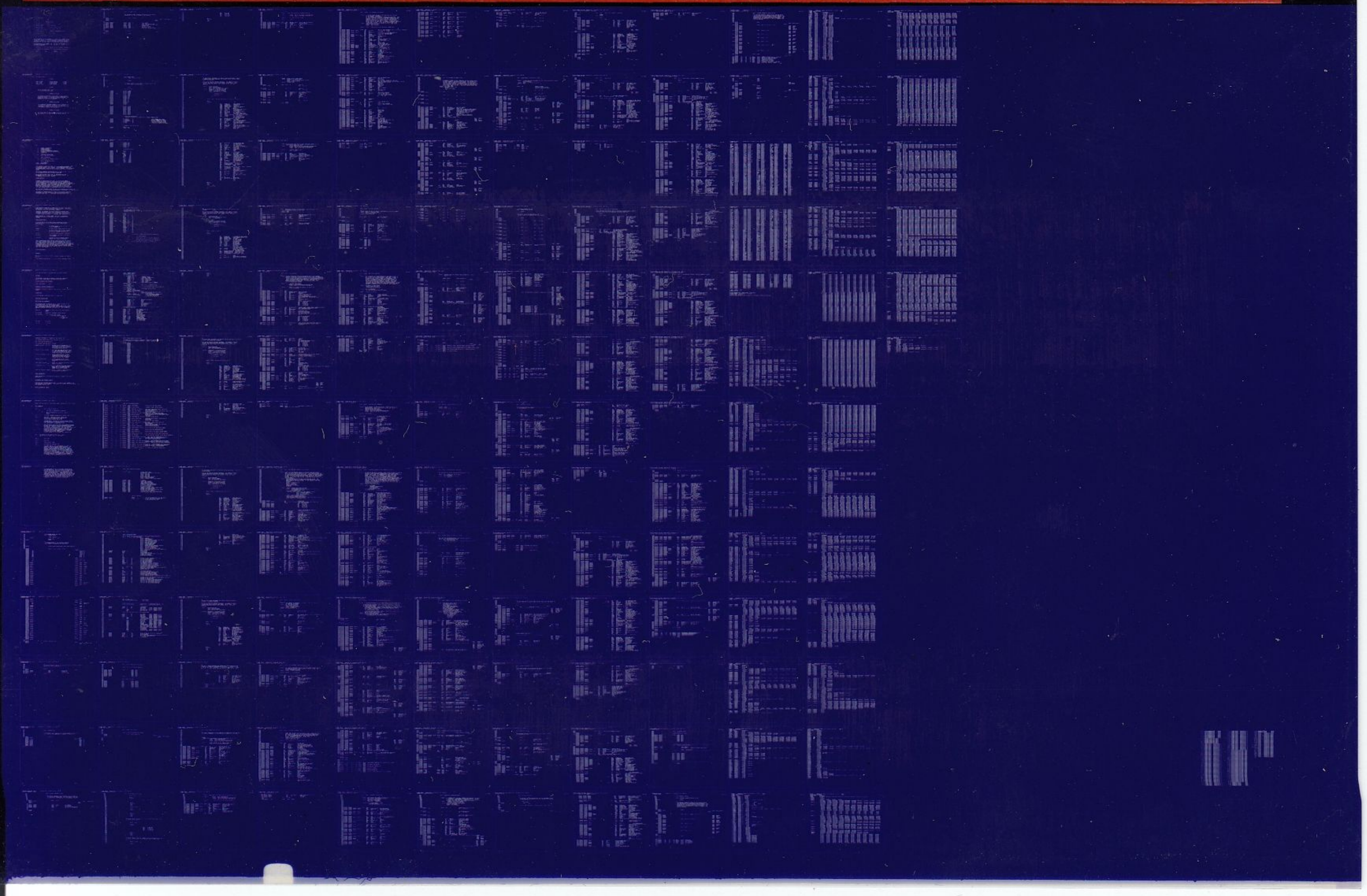
TU58 TU58 PERF EXERCISER AH-E649C-MC APR 1982
CZTUUCO FICHE 1 OF 1
MADE IN USA



.REM &

IDENTIFICATION

PRODUCT CODE: AC-E648C-MC

PRODUCT NAME: CZTUUC0 TU58 PERF EXER

PRODUCT DATE: SEPT 1981

MAINTAINER: DIAGNOSTIC ENGINEERING GROUP

AUTHOR: R. J. ROSS

&

```
.SBTTL  REVISION HISTORY
```

.REM &

HISTORY

JUNE 18,1979	INITIAL RELEASE	CZTUUA0
JULY 1,1979	SECOND RELEASE	CZTUUB0
JUNE 1,1980	THIRD RELEASE	CZTUUB1
OCTOBER 1,1981	FOURTH RELEASE	CZTUUC0

CZTUUA0

1. INITIAL REALEASE--PERF. EXER. FOR UP TO 8 TU58 CONTROLLERS WITH ONE OR TWO DRIVES EACH.

CHANGES TO CZTUUA0

1. THE PROGRAM WAS MODIFIED TO RUN UNDER THE NEW DIAGNOSTIC SUPERVISOR CHSAA0. AS A RESULT OF THIS CONVERSION, THIS PROGRAM NOW OPERATES IN 8K AND PAPERTAPE DISTRIBUTION REQUIRES ONLY ONE PART AK-E650B-MC.

CHANGES TO CZTUUB0

1. ''CLR @ XMSR(R5)'' HAS BEEN CHANGED TO ''DEC @ XMSR(R5)'' TO ALLEVIATE THE PROBLEM OF DESTROYING ANY PREVIOUSLY SET PROGRAMMABLE SPEED IN THE DLV11-E,F, OR DC319 DLART WHEN THE TU58 INIT SEQUENCE WAS TERMINATED.

CHANGES TO CZTUUB1

1. TEST 8 WAS ADDED TO THE DIAGNOSTICS BECAUSE THE TU58 HAS BEEN UPDATED TO USE MODIFIED RADIAL SERIAL PROTOCOL.

&

.REM &

TABLE OF CONTENTS

1.0 GENERAL INFORMATION

THIS DIAGNOSTIC EXERCISES FROM 1 TO 8 TU58 CONTROLLER BOARDS, EACH OF WHICH MAY SUPPORT 1 OR 2 DRIVES. THE PROGRAM IMPLEMENTS THE 'MAINTENANCE MODE'' SWITCH WITHIN ALL PACKET COMMANDS, THUS RETRIEVING MAXIMUM INFORMATION FROM THE DEVICE UPON CERTAIN DEVICE RECOGNIZED ERRORS.

STATISTICAL SUMMARIES ARE PROVIDED FOR ALL UNITS TESTED. RETRIES ARE PERFORMED ON DATA-RELATED ERROR CONDITIONS.

USE OF LOOP ON ERROR FLAG (:LOE) IS IMPLEMENTED BUT NOT RECOMMENDED FOR USE, SINCE THE LOOPS ARE QUITE LENGTHLY DUE TO COMMUNICATIONS PRCTOCOL OVERHEAD.

1.1 PROGRAM ABSTRACT

IN ORDER TO EXERCISE MULTIPLE UNITS IN AN EFFICIENT MANNER, A SCHEDULING ALGORITHM BUILDS, THEN SENDS THE NEXT COMMUNICATION PACKET (COMMAND OR DATA) FORMULATED BY EXECUTING MACRO CODE WITHIN THE TEST ALGORITHMS. THE USE OF MACROS TO IMPLEMENT THE COMMUNICATIONS PROTOCOL SIMPLIFIES CONTEXT SWITCHING FROM UNIT TO UNIT BY NOT REQUIRING 8 SEPARATE DEVICE STACKS IN ADDITION TO THE SYSTEM STACK. THE TEST CODE RUNS AS A CO-ROUTINE WITH THE SCHEDULER,SO A TEST CODE PROGRAM COUNTER IS MAINTAINED FOR EACH UNIT 'TSTPC(R5)".

THE TESTS ARE PERFORMED USING THE SPECIFIED ALGORITHM ON ALL DRIVE 0'S, THEN REPEAT THE TEST AFTER SWITCHING DRIVES, IF ANY DRIVE ''1'S'' WERE SELECTED.

FOLLOWING THE TRANSMISSION OF 1 PACKET TO EACH DEVICE (WITH XOFF PRECEEDING) THE UNITS ARE POLLED, AND THEIR ENTIRE RESPONSES EVALUATED ROUND ROBIN. IF ANY ERROR INITIATES A RETRY, THE SCHED-

ULING PROCESS IS MODIFIED TO COMMUNICATE WITH ONLY 1 UNIT UNTIL COMPLETION OF THE RETRY PROCEDURE. THEN, A RETRY BY ANOTHER UNIT MAY PROCEED, OR THE SYSTEM CONTINUES NORMALLY.

THROUGHOUT THE PROGRAM, R5 POINTS TO ONE OF 8 POSSIBLE DATA STRUCTURES CONTAINING STATUS, TEST PARAMETERS, AND STATISTICAL INFORMATION FOR THE CURRENT UNIT, CALLED "UNIT'S DATA BLOCK". "START" CLEARS STATISTICS. "RESTART" AND "CONTINUE" DO NOT.

UPON OCCURANCE OF A FATAL ERROR, THAT UNIT IS DESCHEDULED (ABORTED) ALLOWING THE REMAINING (IF ANY) TO PROCEED WITH TESTING.

ERROR DESCRIPTIONS:

AN EXPLANATION OF THE EXTENDED ERROR INFORMATION FOLLOWS. SEE ALSO THE SECTION IN THIS LISTING SUBTITLED "ERROR MESSAGE DESCRIPTIONS".

BLOCK #:	THE RECORD NUMBER (1 PER 512. BYTES) IN LAST COMMAND PACK.
COMMAND:	THE MOST RECENT COMMAND PACKET OP CODE.
EXPCTD:	THE DATA PATTERN USED ON WRITE COMMAND AND FOR DATA COMPARE AFTER READ OP.
SUCCESS:	THE SUCCESS CODE RECEIVED IN END PACKET.
PAK SENT:	TYPE OF PACKET JUST SENT (0 FOR DATA; 1 FOR COMMAND)
FLAG RCVD:	FLAG BYTE OF PACKET CURRENTLY BEING CHECKED, OR 1ST BYTE OF RESPONSE.

SINCE IN MAINTENANCE MODE TU58 WILL SEND A BAD DATA PACK WITH A "DATA CHECK" SUCCESS STATUS IN THE FOLLOWING END PACK, THE HOST WILL, UPON CHECKING THOSE DATA PACK(S), DETERMINE "BAD DATA" IN PACKET ERROR FIRST, THEN INTERPRET THE SUCCESS CODE TO DIFFERENTIATE A COMMUNICATIONS GLITCH (GOOD SUCCESS) VS. TU 'DATA-CHECK' ERROR CODE. THIS WOULD SEEM TO RESULT IN TWO "ERROR" MESSAGES FOR ONE ERROR CONDITION, BUT ONLY THE SECOND ERROR MESSAGE WILL CONTAIN PERTINENT (NOT ZERO) ERROR NUMBER.

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE

PDP-11/LSI-11 CPU WITH AT LEAST 16K WORDS OF MEMORY AND CONSOLE DEVICE.

TU58 CONTROLLER AND DRIVE(S). DL, DLV, OR PDT COMPATABLE INTER-

FACE; AND REVISION "I" TU58 MICROCODE (OR LATER) ASSUMED.

1.2.2 SOFTWARE

THE PROGRAM IS REVISION D DIAGNOSTIC SUPERVISOR COMPATIBLE.
CONSULT XXDP+ USERS MANUAL FOR OPERATING INSTRUCTIONS.

1.3 RELATED DOCUMENTS AND STANDARDS

XXDP+ USERS MANUAL CHQUS

1.4 DIAGNOSTIC HIERARCY PREREQUISITES

APPROPRIATE INTERFACE DIAGNOSTICS MAY BE RUN TO ISOLATE INTERFACE ERRORS.

1.5 ASSUMPTIONS

SYSTEM HARDWARE OTHER THAN TU58(S) IS OPERATIONAL.

2.0 OPERATING INSTRUCTIONS

2.1 HOW TO RUN THIS DIAGNOSTIC

THE DIAGNOSTIC MAY BE INVOLKED WITH A 'START' RESPONSE TO THE SUPERVISOR PROMPT. 'STA'(CR) IS SUFFICIENT.
IF THE DEVICE IS NOT AT THE STANDARD ADDRESS AND VECTOR (176500, 300), THEN ANSWER "CHANGE HW?" WITH 'YES' INITIALLY TO SET UP HARDWARE CONFIGURATION TABLES FOR EACH UNIT. THAT INFORMATION IS:

TU58 CSR - ADDRESS OF RCSR OF DLV-11 OR OTHER INTERFACE BOARD.

VECTOR ADDR. - ADDRESS OF INTERRUPT VECTOR LOCATION.

PDT INTERFACE -- IS THE TU58 IN A PDT 11/130,
OR SYSTEM WHOSE BUFFERS ARE:
RCSR
RCDB (AND XMDB)
XMSR

TEST DR0 - YES OR NO

TEST DR1 - YES OR NO

SUBSEQUENT RESPONSES TO ''CHANGE HW?'' MAY THEN BE 'NO''.

THE STANDARD ADDRESS AND VECTOR LOCATIONS FOR THE PDT 11/130

ARE 177170 AND 260 RESPECTIVELY.

THE SOFTWARE QUESTIONS ARE AS FOLLOWS:

NUMBER OF BLOCKS: TEST 4-7 -- ONE MAY SELECT A MINIMUM OF 8, TO A MAXIMUM OF 512 BLOCKS TO WRITE, READ; WRITE VERIFY; AND READ REDUCED, AS EXPLAINED IN SECTION 6.0.

ADD DR # TO DATA PATTERN -- FOR THOSE SAME READ AND WRITE TESTS 4-7, THE DRIVE NUMBER (0 OR 1) MAY BE ADDED TO DATA WRITTEN ON TAPE TO INSURE DRIVE SELECT BIT OPERATION.

STATISTICS PRINTED AT EOP -- SELECTS WHETHER OR NOT TO PRINT INFORMATION AT END OF PASS OR ^C. THESE STATISTICS MAY ALSO BE RE-TRIEVED WITH THE 'PRI'' COMMAND.

COMPARE DATA ON READ -- SELECTS WHETHER OR NOT TO DO A DATA COMPARE ON DATA PACKETS RE-CEIVED.

PRINT PACKET ON ERROR -- PRINTS 132. BYTE DATA PACKET ON A COMPARE ERROR,IF SELECTED.

ERRORS=DVC FATAL IF 'EVL' SET -- IF USER SETS EVL FLAG (EVALUATE) MODE), HRD OR SFT ERROR MESSAGES BECOME DVC FTL ERRORS AFTER THE NUMBER SPECIFIED IS EXCEEDED.

PRINT UNIT PROTOCOL SUMMARY (TEST 8) -- PRINTS A TABLE INDICATING THE PROTOCOL OF EACH UNIT.

3.0 ERROR INFORMATION

ERROR INFORMATION IS PROVIDED ON OCCURRANCE OF ERRORS AS OUTLINED IN SECTION 1.1.

4.0 PERFORMANCE AND PROGRESS REPORTS

STATISTICS ARE AVAILABLE PER SECTION 1.1 AT END OF PASS, CONTROL-C, OR UPON ENTERING A 'PRI'' COMMAND. THEY CONSIST OF # BLOCKS WRITTEN AND READ, # OF DATA ERRORS, HARD OR SOFT.

5.0 DEVICE INFORMATION TABLES

CONSULT SECTION SUBTITLED "DATA BLOCK FORMAT" FURTHER ON IN THIS LISTING.

6.0 TEST SUMMARIES

INIT: INIT IS SENT TO DEVICE IF:

1. INIT CODE IN SUPERVISOR IS EXECUTED

OR

2. INIT IS REQUESTED BY DEVICE AS A RESULT OF ERROR.

TEST 1: INITIATES FIRMWARE DIAGNOSTICS AT DEVICE LEVEL (SELF TEST)

TEST 2: SEEK TEST. SEEKS BOT ON BOTH TRACKS,THEN VERIFIES 60 IPS OPERATION TO SEEK EOT ON ON BOTH TRACKS,ENDING THEN AT BOT.

TEST 3: PERFORMS WRITE, THEN READ OF ADJACENT BLOCKS AT BOT WITH VARYING DATA, THEN SEEKS HALF WAY INTO REMAINING TAPE AND REPEATS THE ABOVE UNTIL EOT.

TESTS 4-7: READS OR WRITES BLOCK # AS DATA INTO SUCCESSIVE BLOCKS ON TAPE, THE LENGTH OF WHICH IS DETERMINED BY SOFTWARE QUESTION #1: DEFAULT IS SHORT TAPE (8.).MINIMUM, (8.) RESULTS IN TRANSFER OF 8. (OR 4 PER TRACK) 512. BYTE BLOCKS OF DATA PER READ (OR WRITE) OPERATION. THE ALGORITHM SWITCHES TRACKS REGARDLESS OF THE NUMBER BLOCKS SELECTED. DRIVE NUMBER IS ADDED TO RECORD AS DEFAULT,SO FOR TAPE INTERCHANGE TESTING, ANSWER (N) TO SOFTWARE (SW) QUESTION #2.

NOTE: THE AMOUNT OF TIME SPENT IN TESTS 4-7 IS QUITE LONG IF THE FULL TAPE (512.) IS SELECTED.

TEST 4: WRITE TAPE

TEST 5: READ TAPE

TEST 6: 'WRITE VERIFY' TAPE

TEST 7: READ MODIFIED THRESHOLD TAPE

TEST 8: THE FIRST PART OF TEST 8 DETERMINES IF A UNIT IS CAPABLE OF MODIFIED RADIAL SERIAL PROTOCOL. THIS PART OF THE TEST IS WRITTEN USING RADIAL SERIAL PROTOCOL, AND DETERMINES THE PROTOCOL OF A UNIT BY SENDING THE TU58 A GET CHARACTERISTICS COMMAND AND MONITORING THE RESPONSE. IF THE TU58 RETURNS AN END PACKET IT IS A MODIFIED UNIT. IF THE TU58 RETURNS A DATA PACKET IT IS A NON-MODIFIED UNIT. NOTE, THE DATA PACKET RETURNED ON A GET CHARACTERISTICS COMMAND IS NOT NORMAL, RATHER IT CONSISTS OF A DATA PACKET THAT IS 28. BYTES PLUS AN END PACKET WHICH IS 14. BYTES.

THE SECOND PART OF TEST 8 TESTS ONLY THOUGHS UNITS
THAT ARE MODIFIED. THIS IS ACHIEVED BY LETTING NON-
MODIFIED UNITS JUMP OVER CODE. IT WAS ASSUMED THAT IF
A UNIT CAN READ,WRITE,ETC... WHEN OPERATING IN RSP,
THEN IT CAN READ,WRITE,ETC... WHEN OPERATING IN MRSP.
THEREFORE ALL THAT HAD TO BE TESTED WAS THE ABILITY OF
MODIFIED UNIT TO BE ABLE TO SEND ONE BYTE AND WAIT FOR
A CONTINUE FROM THE HOST BEFORE SENDING THE NEXT BYTE.
A PROTOCOL SUMMARY OF THE UNITS IS ADVAILABLE BY AN-
SWERING YES (Y) TO SOFTWARE (SW) QUESTION # 5.

&

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

                                        .ENABL  ABS,AMA
002000                                  .       =       2000
                                        .NLIST  BEX
002000                                  BGNMOD

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

002000                                  POINTER BGNRPT,BGNSW,BGNSFT,BGNAU,BGNDU,BGNSETUP

002000                                  HEADER  CZTUU,C,0,3600.,1,PRI07
002000                                                                  L$NAME::
002000  103                                                                     .ASCII /C/
002001  132                                                                     .ASCII /Z/
002002  124                                                                     .ASCII /T/
002003  125                                                                     .ASCII /U/
002004  125                                                                     .ASCII /U/
002005  000                                                                     .BYTE   0
002006  000                                                                     .BYTE   0
002007  000                                                                     .BYTE   0
002010                                                                  L$REV::
002010  103                                                                     .ASCII  /C/
002011                                                                  L$DEPO::
002011  060                                                                     .ASCII  /0/
002012                                                                  L$UNIT::
002012  000001                                                                  .WORD   T$PTHV
002014                                                                  L$TIML::
002014  007020                                                                  .WORD   3600.
002016                                                                  L$HPCP::
002016  037614                                                                  .WORD   L$HARD
002020                                                                  L$SPCP::
002020  037756                                                                  .WORD   L$SOFT
002022                                                                  L$HPTP::
002022  002174                                                                  .WORD   L$HW
002024                                                                  L$SPTP::
002024  002206                                                                  .WORD   L$SW
002026                                                                  L$LADP::
002026  040422                                                                  .WORD   L$LAST
002030                                                                  L$STA::
002030  000000                                                                  .WORD   0
002032                                                                  L$CO::
002032  000000                                                                  .WORD   0
002034                                                                  L$DTYP::
002034  000001                                                                  .WORD   1
002036                                                                  L$APT::
002036  000000                                                                  .WORD   0
002040                                                                  L$DTP::
002040  002152                                                                  .WORD   L$DISPATCH
002042                                                                  L$PRIO::
002042  000340                                                                  .WORD   PRI07
002044                                                                  L$ENVI::
```

```
002044  000000                                          .WORD   0
002046                                          L$EXP1::
002046  000000                                          .WORD   0
002050                                          L$MREV::
002050     003                                          .BYTE   C$REVISION
002051     003                                          .BYTE   C$EDIT
002052                                          L$EF::
002052  000000                                          .WORD   0
002054  000000                                          .WORD   0
002056                                          L$SPC::
002056  000000                                          .WORD   0
002060                                          L$DEVP::
002060  005506                                          .WORD   L$DVTYP
002062                                          L$REPP::
002062  015014                                          .WORD   L$RPT
002064                                          L$EXP4::
002064  000000                                          .WORD   0
002066                                          L$EXP5::
002066  000000                                          .WORD   0
002070                                          L$AUT::
002070  017146                                          .WORD   L$AU
002072                                          L$DUT::
002072  017022                                          .WORD   L$DU
002074                                          L$LUN::
002074  000000                                          .WORD   0
002076                                          L$DESP::
002076  002122                                          .WORD   L$DESC
002100                                          L$LOAD::
002100  104035                                          EMT     E$LOAD
002102                                          L$ETP::
002102  000000                                          .WORD   0
002104                                          L$ICP::
002104  016030                                          .WORD   L$INIT
002106                                          L$CCP::
002106  017002                                          .WORD   L$CLEAN
002110                                          L$ACP::
002110  016620                                          .WORD   L$AUTO
002112                                          L$PRT::
002112  002142                                          .WORD   L$PROT
002114                                          L$TEST::
002114  000000                                          .WORD   0
002116                                          L$DL":::
002116  000000                                          .WORD   0
002120                                          L$HIME::
002120  000000                                          .WORD   0
495
496 002122                      DESCRIP <TU58 PERF EXER>
002122                                          L$DESC::
002122     124     125     065                          .ASCIZ  /TU58 PERF EXER/
                                                        .EVEN
```

```
                        ;++
                        ;THE PROTECT TABLE IS USED BY THE MONITOR TO WARN THE OPERATOR WHEN HE
                        ;TRIES TO TEST THE LOAD DEVICE.
                        ;--

002142                  BGNPROT
002142                                                          L$PROT::
002142  000000                .WORD   0               ;DEVICE CSR
002144  177777                .WORD   -1              ;NO MASS BUS
002146  177777                .WORD   -1              ;NO DRIVE
002150                  ENDPROT
```

```
                                .SBTTL  DISPATCH TABLE

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

002150                                  DISPATCH 8
002150  000010                                                         .WORD   8
002152                                                          L$DISPATCH::
002152  017150                                                          .WORD   T1
002154  017352                                                          .WORD   T2
002156  017624                                                          .WORD   T3
002160  021230                                                          .WORD   T4
002162  022220                                                          .WORD   T5
002164  023004                                                          .WORD   T6
002166  023774                                                          .WORD   T7
002170  024560                                                          .WORD   T8
```

```
                                .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.
                                ;--

002172                                  BGNHW   DFPTBL
002172  000004                                                         .WORD   L10001-L$HW/2
002174                                                          L$HW::
002174                                                          DFPTBL::

002174  176500                  .WORD   176500          ;CSR ADDRESS
002176  000300                  .WORD   300             ;VECTOR ADDR.
002200  000003                  .WORD   3               ;TEST DRIVE ZERO AND ONE
002202  000000                  .WORD   0               ;NOT PDT TYPE INTERFACE

002204                                  ENDHW
002204                                                          L10001:
```

```
                                .SBTTL  SOFTWARE P-TABLE

                                ;++
                                ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
                                ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
                                ;--

002204                                  BGNSW   SFPTBL
002204  000007                                                          .WORD   L10002-L$SW/2
002206                                                                  L$SW::
002206                                                                  SFPTBL::

002206  000010                  LENGTH: .WORD   8.              ;TAPE LENGTH
002210  000001                  STAEOP: .WORD   1               ;PRINT STATISTICS AT EOP
002212  000001                  PRBUF:  .WORD   1               ;PRINT DATA BUF ON COMP. ERROR
002214  000001                  CMPDAT: .WORD   1               ;COMPARE DATA
002216  000001                  DRVCHK: .WORD   1               ;ADD DR # TO DATA
002220  000001                  EVLTHR: .WORD   1               ;THRESHOLD FOR EVL TEST
002222  000000                  PPSOT8: .WORD   0               ;PRINT UNIT PROTOCOL SUMMARY (TST8)


002224                                  ENDSW
002224                                                                  L10002:

002224                                  ENDMOD
```

```
                                .TITLE GLOBAL AREAS
                                .SBTTL  GLOBAL EQUATES SECTION
                                
                                
002224                                  BGNMOD
                                
                                ;++
                                ; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
                                ; ARE USED IN MORE THAN ONE TEST.
                                ;--
                                
002224                                  EQUALS
                                ;
                                ; BIT DIFINITIONS
                                ;
100000                          BIT15== 100000
040000                          BIT14== 40000
020000                          BIT13== 20000
010000                          BIT12== 10000
004000                          BIT11== 4000
002000                          BIT10== 2000
001000                          BIT09== 1000
000400                          BIT08== 400
000200                          BIT07== 200
000100                          BIT06== 100
000040                          BIT05== 40
000020                          BIT04== 20
000010                          BIT03== 10
000004                          BIT02== 4
000002                          BIT01== 2
000001                          BIT00== 1
                                ;
001000                          BIT9==  BIT09
000400                          BIT8==  BIT08
000200                          BIT7==  BIT07
000100                          BIT6==  BIT06
000040                          BIT5==  BIT05
000020                          BIT4==  BIT04
000010                          BIT3==  BIT03
000004                          BIT2==  BIT02
000002                          BIT1==  BIT01
000001                          BIT0==  BIT00
                                ;
                                ; EVENT FLAG DEFINITIONS
                                ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                ;
000040                          EF.START==      32.                     ; START COMMAND WAS ISSUED
000037                          EF.RESTART==    31.                     ; RESTART COMMAND WAS ISSUED
000036                          EF.CONTINUE==   30.                     ; CONTINUE COMMAND WAS ISSUED
000035                          EF.NEW==        29.                     ; A NEW PASS HAS BEEN STARTED
000034                          EF.PWR==        28.                     ; A POWER-FAIL/POWER-UP OCCURRED
                                ;
                                ;
                                ; PRIORITY LEVEL DEFINITIONS
                                ;
000340                          PRI07== 340
000300                          PRI06== 300
```

```
                000240                  PRI05== 240
                000200                  PRI04== 200
                000140                  PRI03== 140
                000100                  PRI02== 100
                000040                  PRI01== 40
                000000                  PRI00== 0
                                        ;
                                        ;OPERATOR FLAG BITS
                                        ;
                000004                  EVL==     4
                000010                  LOT==    10
                000020                  ADR==    20
                000040                  IDU==    40
                000100                  ISR==   100
                000200                  UAM==   200
                000400                  BOE==   400
                001000                  PNT==  1000
                002000                  PRI==  2000
                004000                  IXE==  4000
                010000                  IBE== 10000
                020000                  IER== 20000
                040000                  LOE== 40000
                100000                  HOE== 100000
    641
```

```
                        .SBTTL  ERROR CODE EQUATES

                        ;THE ERROR CODE OFFSET VALUES :
                        ;USED BY ROUTINE 'LOG' TO INDEX (BY R5) INTO DEVICE'S DATA BLOCK AND
                        ;INCREMENT STATISTICS.

000002                  SFTRD   ==      2
000004                  SFTWR   ==      4
000006                  RCINIT  ==      6
000010                  OTL     ==      8.
000012                  OVRN    ==      10.
000014                  BDCOM   ==      12.
000016                  HRDRD   ==      14.
000020                  HRDWR   ==      16.
000022                  BDCHK   ==      18.
000024                  SKERR   ==      20.
000026                  WRLOCK  ==      22.
000030                  NOMOT   ==      24.
000032                  CNINIT  ==      26.
000034                  PARTL   ==      28.
000036                  NOUNIT  ==      30.
000040                  CMNDER  ==      32.
000042                  RECERR  ==      34.
000044                  SLFER   ==      36.
000046                  SUCOTL  ==      38.
000050                  TORCVB  ==      40.
000054                  NCART   ==      44.
000056                  TOSNDB  ==      46.

                        ;               IN ADDITION, SYSTEM SETUP OR RUNTIME ERRORS ARE:

                        ;       100.    -       ALL UNITS ABORTED

                        ;       101.    -       MORE THAN 8. UNITS (16 DRIVES) REQUESTED

                        ;       102.    -       NEITHER DRIVE SELECTED FOR THIS CONTROLLER

                        ;               ALL THE ABOVE ARE CLASSIFIED AS SYSTEM FATAL
```

```
                        .SBTTL  GENERAL EQUATES
                        ;RADIAL SERIAL CODES:
                        ;-----------------------------------------------------------------------
                        ;THE FLAG BYTE CODES ARE:
000002                          RSCMND == 2             ;"COMMAND" PACKET
000020                          RSCONT == 20            ;"CONTINUE" SINGLE BYTE
000020                          RSXON  == 20            ;"XON" SINGLE BYTE
000023                          RSXOFF == 23            ;"XOFF" SINGLE BYTE
000004                          RSINIT == 4             ;"INIT" SINGLE BYTE
000001                          RSDATA == 1             ;"DATA" PACKET
000002                          RSEND  == RSCMND        ;"END" PACKET FLAG IS "COMMAND"
                        ;-----------------------------------------------------------------------
                        ;END PACK SIZE:
000016                          RSNDSZ == 14.           ;TOTAL BYTES IN COMMAND PACKET
                        ;MESSAGE PACK SIZE:
000012                          RSMSIZ == 12            ;10. BYTES FOR BYTE COUNT INSIDE CMND PACK
                        ;DATA PACK SIZE:
000204                          RSDASZ == 132.          ;TOTAL BYTES IN DATA PACKET
                        ;DATA + END PACK SIZE:
000222                          RSDNSZ == RSDASZ+RSNDSZ
                        ;GET CHARACTERISTICS DATA PACKET SIZE
000034                          RSGCDP == 28.           ;TOTAL BYTES FOR GET CHAR DATA PACKET
                                                        ;MINUS THE END PACKET
000016                          RSSNSZ =  RSMSIZ + 4            ;SIZE FOR SENDING COMMAND PACK
001036                          RCBFSZ== 4*RSDASZ+RSNDSZ        ;4 DATA PAKS AND END PACK
                                                                ;IS SIZE OF RCV BUFFERS
                        ;-----------------------------------------------------------------------
                        ;       THE OP CODES ARE:

000100                          RSSEND == 100           ;END PACK DESCRIPTOR
000003                          RSSWR  == 3             ;WRITE
000002                          RSSRD == 2              ;READ
000005                          RSSSEK == 5             ;SEEK
000012                          RSSGET == 12            ;GET CHARACTERISTICS
000000                          RSSNOP == 0             ;NO-OPERATION
000001                          RSSNIT == 1             ;INITIALIZE
000007                          RSSSLF == 7             ;SELF TEST
                        ;-----------------------------------------------------------------------
                        ;THE SUCCESS CODES ARE:

177720                  ESABO   ==-48.          ;BAD COMMAND FROM HOST
177767                  ESNCRT  ==-9.           ;NO CARTRIDGE
177770                  ESNONX  ==-8.           ;NO DRIVE
000000                  ESOK    ==0             ;OP COMPLETE SUCCESS
177776                  ESPART  ==-2            ;PARTIAL OP
177740                  ESSK    ==-32.          ;SEEK ERROR
000001                  ESTRY   ==1             ;RETRYS OCCURRED
177765                  ESWLOC  ==-11.          ;WRITE PROTECTED
177737                  ESNOMO  ==-33.          ;MOTOR STOPPED
177720                  ESCMD   ==-48.          ;COMMAND ERROR
177711                  ESREC   ==-55.          ;BAD RECORD NUMBER.
177757                  ESCKS   ==-17.          ;TU CHKSUM ERROR
177777                  ESSLF   ==-1.           ;SELF TEST ERROR
177757                  ESCKSM=ESCKS
177757                  ESWR=ESCKS
177757                  ESRD=ESCKS
                        ;-----------------------------------------------------------------------
```

```
     752                            .SBTTL  ERROR MESSAGE DESCRIPTIONS
     753
     754                            ;THE TABLE OF ERROR MESSAGES (ADDRESSES).  ABNDX(R5) CONTAINS THE OFFSET
     755                            ;OF THE REASON.  IT'S ABSOLUTE ADDRESS IS RSNTAB + ABNDX(R5).
     756
     757 002224  002320             RSNTAB: MSNLOG
     758 002226  003052                     MSSFRD
     759 002230  003112                     MSSFWR
     760 002232  002534                     MSRNIT
     761 002234  002776                     MSQRSP
     762 002236  003256                     MSOVRN
     763 002240  002402                     MSCOM
     764 002242  003152                     MSHDRD
     765 002244  003214                     MSHDWR
     766 002246  002554                     MSHCHK
     767 002250  002304                     MSSKER
     768 002252  002512                     MSWPRO
     769 002254  002444                     MSNOMO
     770 002256  002616                     MSNIT
     771 002260  002632                     MSPART
     772 002262  002654                     MSUNIT
     773 002264  002702                     MSCMD
     774 002266  002716                     MSREC
     775 002270  002362                     MSSELF
     776 002272  002736                     MSWRSP
     777 002274  002762                     MSNRSP
     778 002276  002320                     MSNLOG
     779 002300  002462                     MSNOTP
     780 002302  003030                     MSTOSN
```

```
                        ;HERE ARE THE MESSAGES PROPER:

002304  123  105  105  MSSKER:: .ASCIZ /SEEK ERROR/            ;DEVICE COULD NOT READ HEADER
                        .EVEN
002320  123  131  123  MSNLOG:: .ASCIZ /SYSTEM ERROR/          ;DIAGNOSTIC HUNG. BETTER RE-BOOT
                        .EVEN
002336  102  101  104  MSBDA:: .ASCIZ  /BAD DATA IN PACKET/    ;HOST DATA CHECK FOUND ERROR, DEVICE MAY
                        .EVEN                                  ;HAVE READ CORRECTLY.
002362  123  105  114  MSSELF:: .ASCIZ /SELF TEST ERROR/       ;MICRO DIAGNOSTIC FAILED, BUT DEVICE COULD STILL
                        .EVEN                                  ;SEND AN END PACKET.
002402  102  101  104  MSCOM:: .ASCIZ  /BAD DATA W-O DATA CHECK ERR AT TU/  ;PREVIOUS DATA CHECK
                        .EVEN                                  ;ERROR NOT DUE TO DEVICE READ OPERATION
002444  115  117  124  MSNOMO:: .ASCIZ /MOTOR STOPPED/         ;DEVICE COULD NOT GET ANY MEANINGFUL SIGNAL
                        .EVEN                                  ;FROM TAPE.
002462  103  101  122  MSNOTP:: .ASCIZ /CARTRIDGE NOT IN PLACE/ ;NO MEDIA OR BAD SWITCH
                        .EVEN
002512  127  122  111  MSWPRO:: .ASCIZ /WRITE PROTECTION/       ;CARTRIDGE WRITE PROTECT TAB MISSING OR
                        .EVEN                                  ;SWITCH BAD
002534  122  105  103  MSRNIT:: .ASCIZ /RECIEVING INIT/         ;DEVICE SENT INIT REQUEST
                        .EVEN
002554  110  117  123  MSHCHK:: .ASCIZ /HOST FOUND PACKET CHECKSUM ERROR/  ;DEVICE SENT PACK WITH
                        .EVEN                                  ;BAD CHECKSUM
002616  103  101  116  MSNIT:: .ASCIZ  /CAN'T INIT/             ;DEVICE SENT BYTE OTHER THAN ''CONTINUE''
                        .EVEN                                  ;DURING INITIALIZATION
002632  120  101  122  MSPART:: .ASCIZ /PARTIAL OPERATION/      ;END OF MEDIUM ENCOUNTERED
                        .EVEN
002654  042  116  117  MSUNIT:: .ASCIZ /''NON-EXISTENT'' DRIVE/  ;DEVICE RECV'D TOO LARGE DRIVE NUMBER
                        .EVEN
002702  102  101  104  MSCMD:: .ASCIZ  /BAD COMMAND/            ;DEVICE COULD NOT UNDERSTAND HOST
                        .EVEN
002716  102  101  104  MSREC:: .ASCIZ  /BAD RECORD NO./         ;DEVICE RECV'D TOO LARGE A RECORD NUMBER
                        .EVEN
002736  127  122  117  MSWRSP:: .ASCIZ /WRONG SUCCESS CODE/     ;HOST COULD NOT DECIPHER CODE IN END PACK
                        .EVEN
002762  116  117  040  MSNRSP:: .ASCIZ /NO RESPONSE/           ;TIME OUT WAITING FOR BYTE IN RCV BUF ON INTERFACE.
                        .EVEN
002776  111  116  104  MSQRSP:: .ASCIZ \INDECIPHERABLE FLAG BYTE\  ;HOST COULD NOT UNDERSTAND 1ST BYTE OF
                        .EVEN                                  ;RESPONSE FROM TU AS PROPER PROTOCOL
003030  124  111  115  MSTOSN:: .ASCIZ /TIME OUT ON SEND/       ;DLV 'READY' NEVER WENT HIGH
                        .EVEN
003052  122  105  103  MSSFRD:: .ASCIZ /RECOV. DATA CHECK ERR ON RD OP/  ;TU58 RESPONDED WITH 'DATA-CHECK''
                        .EVEN                                  ;ERROR ON READ OP. ;HOST RETRY(S) SUCCESSFUL
003112  122  105  103  MSSFWR:: .ASCIZ /RECOV. DATA CHECK ERR ON WR OP/  ;SAME BUT WR OR WR VERIFY OPERATION
                        .EVEN
003152  125  116  122  MSHDRD:: .ASCIZ /UNRECOV. DATA CHECK ERR ON RD OP/  ;TU58 RESPONDED WITH 'DATA-CHECK''
                        .EVEN                                  ;ERROR ON READ OP.  ;RETRIES UNSUCCESSFUL
003214  125  116  122  MSHDWR:: .ASCIZ /UNRECOV. DATA CHECK ERR ON WR OP/  ;SAME BUT WR OPERATION
                        .EVEN
003256  104  114  126  MSOVRN:: .ASCIZ /DLV ERROR IN RECEIVE/  ;DLV ERROR (THE CONTENTS PRINTED OUT)
                        .EVEN
```

```
                                .SBTTL  MISC STORAGE AND EQUATES

003304  000000          SYSTAT:: .WORD                  ;SYSTEM STATUS WORD
                                                        ;BIT7-BIT15 = 1ST BYTE OF PACK RCVD
                                                        ;BIT05 = NOT USED
                                                        ;BIT04 = NOT USED
                                                        ;BIT03 = NOT USED
                                                        ;BIT02 =RETRY BAD FLAG BYTE
                                                        ;BIT01 = RETRY OCCURRING
                                                        ;BIT00 = CHKSUM IS ODD (TEMP STORAGE)

003306  000000          TAPLEN:: .WORD                  ;# RECORDS
003310  000000          DEVPTR:: .WORD                  ;->NEXT UNIT ADDRESS
003312  000000          RCFLG::  .WORD                  ;TEMP STORE FOR GTBYTE
003314  000000          RCBCNT:  .WORD                  ;TEMP STORAGE FOR GTBYTE
003316  000004          FTLNM:   .WORD 4                ;NUMBER OF TRIES BEFORE ABORT
003320  000000          DONE::   .WORD                  ;1 IF TEST ALGORITHM COMPLETE
003322  000000          IDPTR::  .WORD                  ;PTR IN TESTID MACRO
003324  000000          TSTTOP:  .WORD                  ;POINTER TO CURRENT TEST
003326  000010          MXRTRY:: .WORD  8.              ;NUMBER OF RETRIES, BEFORE "HARD" ERROR
003330  000000          BLKER::  .WORD                  ;RECORD # FOR ERROR REPORT
003332  000000          SECREC:: .WORD                  ;RECORD # TO START AFTER SWITCHING TRACKS
003334  000000          PRNSIZ:: .WORD                  ;SIZE OF PACK FOR "PRNPAK"
003336  000000          ALLGON:: .WORD                  ;SECONDARY DON'T PRINT STATISTICS FLAG.
003340  000000          TEST8::  .WORD                  ;***** FLAG THAT INDICATES IF IN TST 8


                        ;               TIME OUT CONSTANTS:

003342  177777                           CSNRDY::  -1           ;TIME OUT ON SEND DELAY (DLV NOT READY)
003344  000226                           CSRCVB::  150.         ;45 SEC. MULTIPLIER  REWIND TIME
                                                                ;CAN BE OPTIMISED PER CPU
```

```
                        .SBTTL  DATA BLOCK FORMAT
                        ;-------------------------------------------------------------------------
                        ;R5 --> TOP OF 1 OF THE 8 DATA BLOCKS (1 PER UNIT) DURING EXECUTION
                        ;@R5 IS THE STATUS WORD CONTAINING:
                                                        ;BIT15 = ABORTED
                                                        ;BIT14 = SEND "BREAK"
                                                        ;BIT13 = RETRY FLAG BYTE ERROR (DATA PACKS)
                                                        ;BIT12 = TEMP STOR WRITE MACRO
                                                        ;BIT11 = UNIT NOT BEING TESTED
                                                        ;BIT10 = RETRYING DATA ERROR
                                                        ;BIT9  = TU58 CHKSUM ERROR
                                                        ;BIT8  = RD/WR OPERATION
                                                        ;BIT7  = NORMAL/REDUCED THRESHOLD (MACROS)
                                                        ;BIT6  = HOST DATA COMPARE ERROR
                                                        ;BIT5  = WR VERIFY OPERATION
                                                        ;BIT4  = TYPE OF PAK SENT 0DATA 1CMD
                                                        ;BIT3  = RETRY FLAG BYTE ERR.(SEND COMMAND PACK)
                                                        ;BIT0,1,2=UNIT NO.
000000                  STATUS  ==      0.              ;DEVICE STATE
000002                  RETRY   ==      2.              ;# OF RETRIES
000004                  ABNDX   ==      4.              ;ERROR NUMBER FOR LOG
                        ;R0                             ;STORAGE FOR REGISTERS USED IN TEST BODY
                        ;R1                             ;STORED WITH SWAPOW
                        ;R2                             ;RETRIEVED WITH SWAPIN
                        ;R3                             ;
                        ;R4                             ;
000020                  TSTPC   ==      16.             ;POINTER TO NEXT EXECUTABLE TEST INST.
000022                  RCSR    ==      18.             ;DLV RCV STATUS ADDRESS
000024                  RCDB    ==      20.             ;DLV RCV DATA ADDRESS
000026                  XMSR    ==      22.             ;DLV SND STATUS ADDRESS
000030                  XMDB    ==      24.             ;DLV SND DATA ADDRESS
000032                  XSPKNM  ==      26.             ;THE NUMBER OF PACKETS TO RECEIVE
000034                  XSFLG   ==      28.             ;THE EXPECTED FLAG OF 1ST PACKET
000036                  XSCNT   ==      30.             ;THE EXPECTED COUNT OF 1ST PACKET
                                                        ;FOR MULTIPLE PACKET RECIEVES (MAX.4)
                                .       BLKW    8.      ;CONSECUTIVE XSFLGS AND XSCNTS
000060                  DR      ==      48.             ;DR==0 OR 1; BIT8,9 DRIVE SELECTED BY OPERATOR
000062                  TRK     ==      50.             ;COUNTER FOR TRACK NUMBER
000064                  REC     ==      52.             ;RECORD (BLOCK #)

000066                  TMP     ==      54.             ;TEST MACRO REGISTER
000070                  SNDCNT  ==      56.             ;THE # OF BYTES FOR SENDING PACKET
000072                  PATTEN  ==      58.             ;DATA PATTERN-LOWER BYTE USED
000074                  DLV     ==      60.             ;CONTENTS OF RCDB ON DLV ERROR
000076                  SUCCS   ==      62.             ;SUCCESS CODE OF LAST END PACKET
000100                  CMDSNT  ==      64.             ;TYPE OF COMMAND CURRENT IN EVEN BYTE; BIT15==VERIFY OP.

000102                  RCVBUF  ==      66.             ;POINTER TO 542. BYTE BUFFER (4 DATA PAKS + END PACK)
000104                  PKPTR   ==      68.             ;POINTER TO TOP OF PACKET
000106                  XSPTR   ==      70.             ;POINTER TO CURRENTLY USED XSFLG OR XSCNT
000110                  WRTN0   ==      72.             ;THE # OF 512. BYTE BLOCKS WRITTEN DR0
000112                  WRTN1   ==      74.             ;THE # OF 512. BYTE BLOCKS WRITTEN DR1
000114                  RDN0    ==      76.             ;THE # OF 512. BYTE BLOCKS READ DR0
000116                  RDN1    ==      78.             ;THE # OF 512. BYTE BLOCKS READ DR1
```

```
                        ;AND THE ERROR LOG...                   +-----------+
                        ;SPLIT INTO A BYTE PER DRIVE:           ! DR1 ! DR0 !
                        ;                                       +-----------+
                        ;
                        ;-------------------------------------------------------------------
                        ;OFFSET IN DATA BLOCK          ;ERROR TYPE     ;ERRCODE;MSG CODE;SUC. CODE
                        ;-------------------------------------------------------------------
000120                  LGOFST  ==      80.             ;**RESERVED**
000122                  SOFTR   ==      82.             ;SOFT READ      ;SFTRD  ;MSSFRD ;ESCKSM
000124                  SOFTW   ==      84.             ;SOFT WRITE     ;SFTWR  ;MSSFWR ;ESSKSM
                        ;       WORD                    ;RECIEVED INIT  ;RCINIT ;MSRNIT ;******
                        ;       WORD                    ;BAD FLAG BYTE  ;OTL    ;MSQRSP ;******

                        ;THEN THOSE CODES WHICH HAVE N TRIES BEFORE ABORT

000132                  T4TRY   ==      90.             ;DLV ERROR      ;OVRN   ;MSOVRN ;******
000134                  BDATA   ==      92.             ;BAD DATA       ;BDCOM  ;MSDATA ;******
000136                  HARDR   ==      94.             ;HARD READ      ;HRDRD  ;MSHDRD ;ESCKSM
000140                  HARDW   ==      96.             ;HARD WRITE     ;HRDWR  ;MSHDWR ;ESCKSM
                        ;       WORD                    ; CHKSM AT HOST ;BDCHK  ;MSHCHK ;******
                        ;       WORD                    ;SEEK ERROR TOTAL;SKERR ;MSSKER ;******
000146                  T1TRY   ==      102.            ;WRITE PROTECT  ;WRLOCK ;MSWPRO ;ESWLOC
                        ;       WORD                    ;NO MOTOR       ;NOMOT  ;MSNOMO ;ESNOMO
                        ;       WORD                    ;CANT INIT      ;CNINIT ;MSNIT  ;******
                        ;       WORD                    ;PARTIAL OP     ;PARTL  ;MSPART ;ESPART
                        ;       WORD                    ;NO UNIT        ;NOUNIT ;MSUNIT ;ESNONX
                        ;       WORD                    ;COMMAND ERROR  ;CMNDER ;MSCMD  ;ESCMD
                        ;       WORD                    ;BAD RECORD NO.;RECERR  ;MSREC  ;ESREC
                        ;       WORD                    ;SELF TEST ERROR;SLFER  ;MSSELF ;******
                        ;       WORD                    ;WRONG SUC.CODE ;SUCOTL ;MSWRSP ;******
                        ;       WORD                    ;NO RESPONSE    ;TORCVB ;MSNRSP ;******
                        ;       WORD                    ;**RESERVED**
                        ;       WORD                    ;NO CARTRIDGE   ;NOCART ;MSNOTP ;ESNCRT
                        ;       WORD                    ;TIME OUT  SEND;TOSNDB  ;MSTOSN ;******


000202                  BLKEND  ==      130.            ;OFFSET OF END OF STATISTICS (RESERVED)
000204                  TUVECT  ==      132.            ;VECTOR ADDRESS
000206                  SAVCNT  ==       134.           ;BYTE COUNT SAVED DURING RETRY ON WRITE OPERATION
000210                  MRSP    ==      136.            ;***** FLAG INDICATING MRSP
000212                  BLKSIZ  ==      138.            ;** RESERVED **
                        ;-------------------------------------------------------------------
```

```
                                .SBTTL  DEVICE DATA BLOCK ALLOCATION


                                ;TABLE OF DEVICE DATA BLOCK ADDRESSES


003346  003366                  BLKTBL::        .WORD   DEV0
003350  003600                                  .WORD   DEV1
003352  004012                                  .WORD   DEV2
003354  004224                                  .WORD   DEV3
003356  004436                                  .WORD   DEV4
003360  004650                                  .WORD   DEV5
003362  005062                                  .WORD   DEV6
003364  005274                  LSTDEV::        .WORD   DEV7


                                ;AND STORAGE FOR EACH:

003366                          DEV0:           .BLKB   BLKSIZ
003600                          DEV1:           .BLKB   BLKSIZ
004012                          DEV2:           .BLKB   BLKSIZ
004224                          DEV3:           .BLKB   BLKSIZ
004436                          DEV4:           .BLKB   BLKSIZ
004650                          DEV5:           .BLKB   BLKSIZ
005062                          DEV6:           .BLKB   BLKSIZ
005274                          DEV7:           .BLKB   BLKSIZ
```

```
                                .SBTTL  GLOBAL TEXT SECTION

                                ;
                                ; NAMES OF DEVICES SUPPORTED BY PROGRAM
                                ;
005506                                  DEVTYP  <TU58 CONTROLLER>
005506                                                          L$DVTYP::
005506  124  125  065                                                .ASCIZ  /TU58 CONTROLLER/
                                                                   .EVEN
```

```
                        .SBTTL  SYSTEM MACRO DEFINITIONS

                                .MACRO  PUSH    ,REG

                                .NLIST
                                .LIST ME
                                .LIST
                                                MOV     REG,-(SP)


                                .NLIST
                                .NLIST ME
                                .LIST
                                .ENDM

                                .MACRO  POP,REG

                                .NLIST
                                .LIST ME
                                .LIST
                                                MOV     (SP)+,REG

                                .NLIST
                                .NLIST ME
                                .LIST
                                .ENDM

                        ;++
                        ;THE MACRO 'SWAPIN' RETRIEVES THE TEST REGISTERS WHICH WERE SAVED
                        ;IN THE DEVICE DATA BLOCK.
                        ;--

                                .MACRO  SWAPIN

                                .NLIST
                                .LIST ME
                                .LIST
                                                MOV     6.(R5),R0
                                                MOV     8.(R5),R1
                                                MOV     10.(R5),R2
                                                MOV     12.(R5),R3
                                                MOV     14.(R5),R4

                                .NLIST
                                .NLIST ME
                                .LIST
                                .ENDM

                        ;++
                        ;THE MACRO 'SWAPOW' SAVES THE CURRENT STATE OF THE UNIT IN THE DRIVE
                        ;DATA BLOCK IN SO THAT THE SCHEDULER MAY 'SWAPIN' ANOTHER UNIT.
                        ;--

                                .MACRO  SWAPOW

                                .NLIST
                                .LIST ME
```

```
GLOBAL AREAS    MACRO M1113  25-SEP-81 10:06  PAGE 23-1
SYSTEM MACRO DEFINITIONS                                                SEQ 0027

    1112                                        .LIST
    1113                                                    MOV     R0,6.(R5)
    1114                                                    MOV     R1,8.(R5)
    1115                                                    MOV     R2,10.(R5)
    1116                                                    MOV     R3,12.(R5)
    1117                                                    MOV     R4,14.(R5)
    1118
    1119                                        .NLIST
    1120                                        .NLIST ME
    1121                                        .LIST
    1122                                        .ENDM
```

```
1125                            ;++
1126                            ;THE WRITE MACRO IMPLEMENTS THE COMPLETE PROTOCOL NECESSARY TO BUILD
1127                            ;A COMMAND PACKET AND SUBSEQUENT DATA PACKETS (UNTIL THE BYTE COUNT
1128                            ;(BCNT) IS SATISFIED).
1129                            ;
1130                            ;SETS UP THE EXPECTED PROTOCOL RESPONSES:  THE NUMBER OF PACKETS
1131                            ;(XSPKNM) AND THEIR FLAG BYTES AND COUNTS (XSFLG, XSCNT).  CALLS
1132                            ;'RSVP' TO SEND EACH PACKET, AND 'CHKSUM' TO CALC. THE PACKET
1133                            ;CHECKSUM.
1134                            ;
1135                            ; INPUTS  - DEVICE BLOCK @R5
1136                            ;           TRBUF - BUFFER ADDRESS
1137                            ;           UNIT'S TEST REGISTERS FROM 'SWAPIN'
1138                            ; OUTPUTS - SNDCNT(R5) = # OF BYTES TO SEND
1139                            ;           XSPKNM = # OF PACKETS EXPECTED
1140                            ;           XSFLG = FLAG BYTE OF 1ST PACKET
1141                            ;           XSCNT = BYTE COUNT OF 1ST PACKET
1142                            ;             . ***
1143                            ;             .   *  SUBSEQUENT XSFLGS
1144                            ;             .    >
1145                            ;             .   *  AND XSCNTS
1146                            ;             . ***
1147                            ;--
1148
1149                                    .MACRO  TUWRIT  PTRN,REC,BCNT,DR,VER,?A,?B,?C,?D,?E,?F,?G,?H,?T
1150
1151                                    .NLIST
1152                                    .LIST ME
1153                                    .LIST
1154                            T:      MOV     #TRBUF,R0       ;MAKE COMMAND PACKET:
1155                                    MOVB    #RSCMND,@R0     ;COMMAND FLAG
1156                                    MOVB    #RSMSIZ,1(R0)   ;THIS SIZE
1157                                    MOVB    #RSSWR,2(R0)    ;INSERT OP CODE-WRITE
1158                                    MOVB    VER,3.(R0)      ;VERIFY (1 OR 0)
1159                                    MOVB    DR,4.(R0)       ;DRIVE #
1160                                    MOVB    #020,5.(R0)     ;MAINTENANCE MODE SWITCH
1161                                    CLR     6.(R0)          ;NO SEQUENCE #
1162                                    MOV     BCNT,8.(R0)     ;TOTAL COUNT TO WRITE
1163                                    MOV     REC,10.(R0)     ;AT  RECORD N
1164                                    MOV     #RSMSIZ,R1      ;THE PACKET SIZE PLUS+2
1165                                    TST     (R1)+           ;(FLAG AND COUNT) INTO R1
1166                                    MOV     #RSSNSZ,SNDCNT(R5)  ;LOAD THE  SIZE TO SEND
1167                                    CALL    CHKSUM          ;R0 --> R1=COUNT
1168                                    MOV     R1,(R0)         ;PUT CHKSUM IN PACKET
1169                                                            ;SET UP EXPECTATIONS:
1170                                    MOV     #RSCONT,XSFLG(R5)   ;THE FLAG
1171                                    MOV     #1,XSCNT(R5)    ;THE COUNT
1172                                    MOV     #1,XSPKNM(R5)   ;THE # PACKETS EXPECTED
1173                                    MOV     BCNT,R2         ;GET # OF DATA BYTES
1174                                    CALL    RSVP            ;SEND (AND RETURN TO SCHEDULER)
1175                                    BIT     #BIT3,@R5       ;FLAG BYTE ERROR?
1176                                    BNE     T               ;YES
1177                                    BIC     #BIT12,@R5      ;FLAG FOR LAST PACKET
1178                            A:      MOV     #TRBUF,R0       ;POINT TO TOP OF BUFFER AGAIN
1179                                    CMP     R2,#128.        ;START DATA PACKET(S)
1180                                    BHI     B               ;BCNT > 128.!
1181                                    MOV     R2,R1           ;BCNT<128.
```

```
1182                                        BIS     #BIT12,@R5      ;SO LAST PACKET NOW
1183                                        BR      C               ;USE REMAINING COUNT
1184                                B:      MOV     #128.,R1        ;USE 128. BYTES
1185                                C:      MOVB    R1,1(R0)        ;COPY COUNT TO BUFFER
1186                                        MOV     R1,R3           ;R3=COUNTER TO LOAD BUFFER
1187                                        MOVB    #RSDATA,@R0     ;FLAG FIRST
1188                                        TST     (R0)+           ;SKIP COUNT
1189                                D:      MOVB    PTRN,(R0)+      ;INSERT DATA
1190                                        DEC     R3              ;MORE?
1191                                        BHI     D               ;YES
1192                                        MOV     #TRBUF,R0       ;-->TOP AGAIN
1193                                        MOVB    1(R0),R1        ;GET COUNT
1194                                        BIC     #177400,R1      ;ZERO SIGN EXTEND
1195                                        MOV     R1,SNDCNT(R5)   ;HOW MANY TO SEND PLUS
1196                                        ADD     #4,SNDCNT(R5)   ;FLAG,COUNT,CHKSUM
1197                                        ADD     #2,R1           ;COMPENSATE FOR FLAG + COUNT
1198                                        CALL    CHKSUM          ;FOR CHECKSUM CALC.
1199                                        MOVB    R1,(R0)+        ;CHKSUM INTO PACKET
1200                                        SWAB    R1              ;EVEN ON AN ODD
1201                                        MOVB    R1,(R0)+        ;BYTE BOUNDARY
1202                                        BIT     #BIT12,@R5      ;LAST DATA PACKET?
1203                                        BEQ     E               ;NO
1204                                        MOV     #RSEND,XSFLG(R5)  ;YES-EXPECT 'END'
1205                                        MOV     #RSNDSZ,XSCNT(R5)  ;OF THIS SIZE
1206                                        MOV     #1,XSPKNM(R5)   ;AND 1 PACKET
1207                                        BR      F               ;SEND
1208                                E:      MOV     #RSCONT,XSFLG(R5)  ;(NOT LAST), EXPECT 'CONTINUE'
1209                                        MOV     #1,XSCNT(R5)    ;AND 1 BYTE
1210                                        MOV     #1,XSPKNM(R5)   ;AND 1 PACKET
1211                                F:      CALL    RSVP            ;SEND PACKET
1212                                                                ;AND RETURN TO SCHEDULER
1213                                        BIT     #BIT3,@R5       ;FLAG BYTE RETRY?
1214                                        BNE     T               ;YES
1215                                        BIT     #BIT10,@R5      ;RETRY DATA ERROR?
1216                                        BNE     G               ;YES
1217                                        SUB     #128.,R2        ;NO, MORE DATA TO SEND?
1218                                        BHI     A               ;YES
1219                                        BR      H               ;NO
1220                                G:      TURTRY  REC,BCNT,DR     ;RETRY HERE
1221                                        BIT     #BIT10!BIT3,@R5 ;RETRY AGAIN?
1222                                        BNE     G               ;YES
1223                                H:      NOP                     ;DONE
1224
1225                            .NLIST
1226                            .NLIST ME
1227                            .LIST
1228                            .ENDM
1229
```

```
                        ;++
                        ;THE SEEK MACRO IMPLIMENTS THE COMPLETE PROTOCOL TO INITIATE A SEEK
                        ;SEQUENCE.
                        ;
                        ;SETS UP THE EXPECTED PROTOCOL RESPONSES:  THE NUMBER OF PACKETS
                        ;(XSPKNM) AND THEIR FLAG BYTES AND COUNTS (XSFLG, XSCNT).  CALLS
                        ;'RSVP' TO SEND EACH PACKET, AND 'CHKSUM' TO CALC. THE PACKET
                        ;CHECKSUM.
                        ;
                        ; INPUTS  - DEVICE BLOCK @R5
                        ;          UNITS TEST REGISTERS FROM SWAPIN
                        ;          TRBUF - BUFFER ADDRESS
                        ;
                        ; OUTPUTS -
                        ;          XSPKNM = # OF PACKETS EXPECTED
                        ;          XSFLG = FLAG BYTE OF 1ST PACKET
                        ;          XSCNT = BYTE COUNT OF 1ST PACKET
                        ;            . ***
                        ;            .   *  SUBSEQUENT XSFLGS
                        ;            .    >
                        ;            .   *  AND XSCNTS
                        ;            . ***
                        ;
                        ;--

                                .MACRO  TUSEEK  REC,DR,?A

                                .NLIST
                                .LIST ME
                                .LIST

A:      MOV     #TRBUF,R0       ;-->(POINT TO) XMIT BUFFER
        MOVB    #RSCMND,@R0     ;FORM COMMAND MESSAGE PACK
        MOVB    #RSMSIZ,1(R0)   ;THIS BIG
        MOVB    #RSSSEK,2(R0)   ;OP CODE IS SEEK
        MOV     REC,10.(R0)     ;TO THIS RECORD
        MOVB    DR,4.(R0)       ;AND WHICH DRIVE
        CLRB    3.(R0)          ;NO MODIFIER
        CLRB    5.(R0)          ;NO SWITCHES
        CLR     6.(R0)          ;NO SEQUENCE #
        CLR     8.(R0)          ;NO BYTE COUNT
        MOV     #RSMSIZ,R1      ;GET COUNT
        TST     (R1)+           ;PLUS FLAG + BCNT
                                ;FOR CHECKSUM CALC
        CALL    CHKSUM          ;R0-->TOP   R1=# OF BYTES
        MOV     R1,(R0)         ;INSERT INTO PACKET
                                ;SET UP EXPECTATIONS:
        MOV     #RSSNSZ,SNDCNT(R5)  ;HOW MANY TO SEND
        MOVB    #RSCMND,XSFLG(R5)  ;EXPECT END PACK
        MOV     #RSNDSZ,XSCNT(R5)  ;COUNT WITH THIS
        MOV     #1.,XSPKNM(R5)  ;EXPECT ONLY 1 PACKET

        CALL    RSVP            ;SEND
                                ;AND RETURN TO SCHEDULER
        BIT     #BIT3,@R5       ;RETRY (FLAG BYTE ERROR)?
        BNE     A               ;YES
```

```
1289                                    .NLIST
1290                                    .NLIST ME
1291                                    .LIST
1292                                    .ENDM
```

```
                        ;++
                        ;THE RETRY MACRO IMPLIMENTS THE COMPLETE PROTOCOL NECESSARY TO INITIATE
                        ;A RETRY (READ OPERATION) SEQUENCE.
                        ;
                        ;SETS UP THE EXPECTED PROTOCOL RESPONSES:  THE NUMBER OF PACKETS
                        ;(XSPKNM) AND THEIR FLAG BYTES AND COUNTS (XSFLG, XSCNT).  CALLS
                        ;'RSVP' TO SEND EACH PACKET, AND 'CHKSUM' TO CALC. THE PACKET
                        ;CHECKSUM.
                        ;
                        ; INPUTS  - DEVICE BLOCK @R5
                        ;           TRBUF - BUFFER ADDRESS
                        ;           UNITS TEST REGISTERS FROM SWAPIN
                        ;
                        ; OUTPUTS - SNDCNT(R5) = # OF BYTES TO SEND
                        ;           XSPKNM = # OF PACKETS EXPECTED
                        ;           XSFLG = FLAG BYTE OF 1ST PACKET
                        ;           XSCNT = BYTE COUNT OF 1ST PACKET
                        ;             . ***
                        ;             .   *  SUBSEQUENT XSFLGS
                        ;             .    >
                        ;             .   *  AND XSCNTS
                        ;             . ***
                        ;--

                                .MACRO  TURTRY  REC,BCNT,DR,?A,?B,?C,?D,?E

                                .NLIST
                                .LIST ME
                                .LIST

                        D:      MOV     #TRBUF,R0       ;FORM CMND PACK:
                                MOVB    #RSCMND,@R0     ;MESSAGE PACK TYPE
                                MOVB    #RSMSIZ,1(R0)   ;THIS BIG
                                MOVB    #RSSRD,2(R0)    ;OP CODE-READ
                                MOV     REC,10.(R0)     ;THIS RECORD
                                MOVB    DR,4.(R0)       ;THIS DRIVE
                                CLRB    3(R0)           ;PRESET NORM THRESHOLD
                                TSTB    @R5             ;REDUCED?
                                BPL     E               ;NO
                                INCB    3(R0)           ;YES-CHANGE THRESHOLD
                        E:      MOV     BCNT,8.(R0)     ;# BYTES DESIRED
                                MOVB    #020,5.(R0)     ;MAINTENANCE MODE
                                CLR     6.(R0)          ;NO SEQUENCE #
                                MOV     #RSMSIZ,R1      ;SIZE OF PACKET
                                TST     (R1)+           ;PLUS FLAG+COUNT INTO R1
                                MOV     #RSSNSZ,SNDCNT(R5)  ;SET UP SIZE TO SEND

                                CALL    CHKSUM          ;FORM CHECKSUM R1=COUNT
                                MOV     R1,(R0)         ;INSERT IN PACKET

                                MOV     BCNT,R1         ;SET EXPECTATIONS:
                                                        ;CALC # OF DATA PACKETS TO EXPECT
                                MOV     #XSFLG,R3       ;OFFSET OF FLAG
                                ADD     R5,R3           ;ABS. ADDR. OF XSFLG
                                CLR     R2              ;PRESET
                        A:      INC     R2              ;# PACKETS EXPECTED
                                MOV     #RSDATA,(R3)+   ;LOAD XSFLG
```

```
1352                                                    MOV     #132.,(R3)+     ;AND EXPECT COUNT
1353                                                    SUB     #128.,R1        ;NEG RESULT LAST TIME
1354                                                    BLOS    C               ;LAST TIME!
1355                                                    BR      A               ;MORE TO DO
1356                                            C:      INC     R2              ;ADD ONE FOR END PACK
1357                                                    MOV     R2,XSPKNM(R5)   ;SAVE # PACKETS TO EXPECT
1358                                                    MOV     #RSEND,(R3)+    ;EXPECT AN END
1359                                                    MOV     #RSNDSZ,(R3)    ;THIS BIG-14. BYTES
1360
1361                                                    CALL    RSVP            ;SEND
1362                                                                            ;AND RETURN TO SCHEDULER
1363
1364                                    .NLIST
1365                                    .NLIST ME
1366                                    .LIST
1367                                    .ENDM
```

```
1370                            ;++
1371                            ;THE READ MACRO IMPLIMENTS THE COMPLETE PROTOCOL NECESSARY TO INITIATE
1372                            ;A READ SEQUENCE.
1373                            ;
1374                            ;SETS UP THE EXPECTED PROTOCOL RESPONSES:  THE NUMBER OF PACKETS
1375                            ;(XSPKNM) AND THEIR FLAG BYTES AND COUNTS (XSFLG, XSCNT).  CALLS
1376                            ;'RSVP' TO SEND EACH PACKET, AND 'CHKSUM' TO CALC. THE PACKET
1377                            ;CHECKSUM.
1378                            ;
1379                            ; INPUTS  - DEVICE BLOCK @R5
1380                            ;           TRBUF - BUFFER ADDRESS
1381                            ;           UNITS TEST REGISTERS FROM SWAPIN
1382                            ;
1383                            ; OUTPUTS - SNDCNT(R5) = # OF BYTES TO SEND
1384                            ;           XSPKNM = # OF PACKETS EXPECTED
1385                            ;           XSFLG = FLAG BYTE OF 1ST PACKET
1386                            ;           XSCNT = BYTE COUNT OF 1ST PACKET
1387                            ;             . ***
1388                            ;             .   *  SUBSEQUENT XSFLGS
1389                            ;             .    >
1390                            ;             .   *  AND XSCNTS
1391                            ;             . ***
1392                            ;--
1393
1394                                    .MACRO  TUREAD  REC,BCNT,DR,VER,?A,?B,?C,?D,?E
1395
1396                                    .NLIST
1397                                    .LIST ME
1398                                    .LIST
1399
1400
1401                            E:      MOV     #TRBUF,R0       ;FORM CMND PACK:
1402                                    MOVB    #RSCMND,@R0     ;MESSAGE PACK TYPE
1403                                    MOVB    #RSMSIZ,1(R0)   ;THIS BIG
1404                                    MOVB    #RSSRD,2(R0)    ;OP CODE IS READ
1405                                    MOV     REC,10.(R0)     ;THIS RECORD
1406                                    MOVB    DR,4.(R0)       ;THIS DRIVE
1407                                    MOVB    VER,3.(R0)      ;VERIFY
1408                                    MOV     BCNT,8.(R0)     ;TOTAL BYTES TO READ
1409                                    MOVB    #020,5.(R0)     ;MAINTENANCE MODE
1410                                    CLR     6.(R0)          ;NO SEQUENCE #
1411                                    MOV     #RSMSIZ,R1      ;GET SIZE OF PACKET
1412                                    TST     (R1)+           ;+2 FOR CHECKSUM
1413                                    MOV     #RSSNSZ,SNDCNT(R5)  ;SIZE TO SEND
1414                                    CALL    CHKSUM          ;FORM CHECKSUM R1=COUNT
1415                                    MOV     R1,(R0)         ;INSERT CHECKSUM
1416
1417                                    MOV     BCNT,R1         ;SET EXPECTATIONS:
1418                                                            ;CALC # OF DATA PACKETS TO EXPECT:
1419                                    MOV     #XSFLG,R3       ;GET OFFSET
1420                                    ADD     R5,R3           ;ABS. ADDR. OF XSFLG
1421                                    CLR     R2              ;PRESET AS NONE
1422                            A:      INC     R2              ;# PACKETS EXPECTED
1423                                    MOV     #RSDATA,(R3)+   ;LOAD XSFLG
1424                                    MOV     #132.,(R3)+     ;AND EXPECTED COUNT
1425                                    SUB     #128.,R1        ;NEG RESULT LAST TIME
1426                                    BLOS    C               ;LAST TIME
```

```
1427                                            BR      A               ;MORE TO DO
1428                                    C:      INC     R2              ;ADD ONE FOR END PACK
1429                                            MOV     R2,XSPKNM(R5)   ;SAVE # PACKETS TO EXPECT
1430                                            MOV     #RSEND,(R3)+    ;EXPECT AN END ALSO...
1431                                            MOV     #RSNDSZ,(R3)    ;THIS BIG-14. BYTES
1432                                            CALL    RSVP            ;SEND
1433                                                                    ;AND RETURN TO SCHEDULER
1434                                    D:      BIT     #BIT10!BIT3,@R5 ;RETRY?
1435                                            BEQ     B               ;NO.
1436                                            TURTRY  REC,BCNT,DR     ;YES
1437                                            BR      D               ;ANOTHER RETRY?
1438                                    B:      NOP                     ;NO
1439
1440
1441                            .NLIST
1442                            .NLIST ME
1443                            .LIST
1444                            .ENDM
```

```
                        ;++
                        ;THE SELF TEST MACRO IMPLIMENTS THE COMPLETE PROTOCOL NECESSARY TO
                        ;INITIATE A 'DIAGNOSE' SEQUENCE.
                        ;
                        ;SETS UP THE EXPECTED PROTOCOL RESPONSES:  THE NUMBER OF PACKETS
                        ;(XSPKNM) AND THEIR FLAG BYTES AND COUNTS (XSFLG, XSCNT).  CALLS
                        ;'RSVP' TO SEND EACH PACKET, AND 'CHKSUM' TO CALC. THE PACKET
                        ;CHECKSUM.
                        ;
                        ; INPUTS  - DEVICE BLOCK @R5
                        ;           TRBUF - BUFFER ADDRESS
                        ;           UNITS REGISTERS TEST FROM SWAPIN
                        ;
                        ; OUTPUTS - SNDCNT(R5) = # OF BYTES TO SEND
                        ;           XSPKNM = # OF PACKETS EXPECTED
                        ;           XSFLG = FLAG BYTE OF 1ST PACKET
                        ;           XSCNT = BYTE COUNT OF 1ST PACKET
                        ;             . ***
                        ;             .   *  SUBSEQUENT XSFLGS
                        ;             .    >
                        ;             .   *  AND XSCNTS
                        ;             . ***
                        ;--

                                .MACRO  TUSELF  ?A

                                .NLIST
                                .LIST ME
                                .LIST

                            A:  MOV     #TRBUF,R0           ;FORM COMMAND PACKET
                                MOVB    #RSCMND,@R0         ;COMMAND FLAG
                                MOVB    #RSMSIZ,1(R0)       ;SIZE OF MESSAGE
                                MOVB    #RSSSLF,2(R0)       ;SELF TEST OPERATION
                                CLRB    3(R0)               ;NO MODIFIER.
                                CLR     4(R0)               ;NO DRIVE OR SWITCHES
                                CLR     6(R0)               ;NO SEQUENCE NUMBER
                                CLR     8.(R0)              ;NO BYTES
                                CLR     10.(R0)             ;NO RECORD #
                                MOV     #RSMSIZ,R1          ;GET SIZE
                                TST     (R1)+               ;+2 FOR CHECKSUM
                                MOV     #RSSNSZ,SNDCNT(R5)  ;SIZE TO SEND
                                CALL    CHKSUM              ;FORM CHECKSUM
                                MOV     R1,(R0)             ;INSERT INTO PACKET
                                MOV     #RSEND,XSFLG(R5)    ;EXPECT END,
                                MOV     #RSNDSZ,XSCNT(R5)   ;THIS BIG
                                MOV     #1,XSPKNM(R5)       ;AND 1 PACKET
                                                            ;SEND
                                CALL    RSVP                ;RETURN TO SCHEDULER
                                BIT     #BIT3,@R5           ;RETRY?(BAD FLAG)
                                BNE     A                   ;YES

                                .NLIST
                                .NLIST ME
                                .LIST
                                .ENDM
```

```
1505                            ;++
1506                            ;THE TEST ID MACRO INTERFACES THE SUPERVISOR'S TEST DISPATCH TO THE
1507                            ;DIAGNOSTIC'S FORMAT BY IMPLEMENTING CALLS THAT:  1) INITIALIZE THE
1508                            ;PC OF THE TEST CODE (TSTPC(R5)),  2) ASSIGN THE 1ST DRIVES,  3) RUN
1509                            ;THE TEST,  4) SWITCH DRIVES AND REINITIALIZE, 5) RUN THE TEST AGAIN.
1510                            ;--
1511
1512                                    .MACRO  TSTID   ADDRES,?A
1513
1514                                    .NLIST
1515                                    .LIST ME
1516                                    .LIST
1517                                                    MOV     ADDRES,TSTTOP   ;SAVE ADDR OF TEST
1518                                                    CALL    SETUP           ;INIT UNITS TSTPC
1519                                                    CALL    SETDR           ;GET 1ST DRVS.
1520                                                    CALL    RUN             ;DO TEST
1521                                                    CALL    SWAPDR          ;GET NEXT DRVS.
1522                                                    BCC     A               ;BR NO 2ND DRVS
1523                                                    CALL    SETUP           ;REINIT UNITS TSTPC
1524                                                    CALL    RUN             ;REPEAT TEST
1525                                            A:                              ;DONE
1526                                    .NLIST
1527                                    .NLIST ME
1528                                    .LIST
1529                                    .ENDM
1530                            ;-----------------------------------------------------------------------
```

```
                                .SBTTL  GLOBAL SUBROUTINES SECTION

                                ;++
                                ; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES THAT ARE USED
                                ; TO LINK THE DIAGNOSTIC TO THE SUPERVISOR (THROUGH THE TSTID MACRO).
                                ;--

                                ;++
                                ; SWAPDR
                                ;    SUBROUTINE TO DETERMINE IF TO TEST OTHER DRIVE (FOR ALL UNITS)
                                ; INPUTS:       DR(R5) - DRIVE CONFIGURATION
                                ;               BLKTBL - TOP OF DATA BLOCK ALLOCATION TABLE
                                ;               LSTDEV - ADDR. OF LAST UNIT'S DATA BLOCK
                                ;
                                ; OUTPUTS:      DR(R5) UPDATED TO TEST SAME OR OTHER DRIVE
                                ;               CARRY SET IF SECOND PASS NECESSARY
                                ;--

005526  005002                  SWAPDR:: CLR    R2              ;FOR # OF DRIVE 1'S.
005530  012737  003346  005626          MOV     #BLKTBL,SWPTR   ;TABLE ADDR. OF 1ST UNIT
005536  017705  000064          1$:     MOV     @SWPTR,R5       ;GET DATA BLOCK ADDR.
005542  032715  100000                  BIT     #BIT15,@R5      ;ABORTED?
005546  001013                          BNE     3$              ;YES
005550  032765  000001  000060          BIT     #BIT0,DR(R5)    ;DID DR. 0?
005556  001007                          BNE     3$              ;NO, DID DR.1 1ST PASS
005560  032765  001000  000060          BIT     #BIT9,DR(R5)    ;YES; 1 SELECTED?
005566  001403                          BEQ     3$              ;NO, ALL DONE
005570  105265  000060                  INCB    DR(R5)          ;YES, SWAP
005574  005202                          INC     R2              ;ONE MORE TO TEST
005576  023727  005626  003364  3$:     CMP     SWPTR,#LSTDEV   ;LAST DEVICE?
005604  103004                          BHIS    4$              ;YES
005606  062737  000002  005626          ADD     #2,SWPTR        ;NO-POINT NEXT
005614  000750                          BR      1$              ;DO

005616  005702                  4$:     TST     R2              ;(CLEAR CARRY),MORE TO DO?
005620  001401                          BEQ     5$              ;NO
005622  000261                          SEC                     ;YES
005624  000207                  5$:     RETURN                  ;RETURN

005626  000000                  SWPTR:  .WORD
```

```
                                ;++
                                ; SETDR - SUBROUTINE TO GET DRIVE FOR 1ST PASS FOR EACH TEST
                                ;
                                ; INPUTS:      DR(R5) - DRIVE CONFIGURATION
                                ;              BLKTBL - TOP OF DATA BLOCK ALLOCATION TABLE
                                ;              LSTDEV - ADDR. OF LAST UNIT'S DATA BLOCK
                                ;
                                ; OUTPUTS:     DR(R5) IS SET TO TEST DRIVE 0 OR DRIVE 1
                                ;--

005630  012737  003346  005704  SETDR:: MOV     #BLKTBL,SETPTR  ;TABLE OF ADDR. 1ST UNIT
005636  017705  000042          1$:     MOV     @SETPTR,R5      ;GET DATA BLOCK ADDR.
005642  105065  000060                  CLRB    DR(R5)          ;PRESET AS DR0
005646  032765  000400  000060          BIT     #BIT8,DR(R5)    ;DO DR0?
005654  001002                          BNE     2$              ;YES
005656  105265  000060                  INCB    DR(R5)          ;NO-USE DRIVE 1
005662  023727  005704  003364  2$:     CMP     SETPTR,#LSTDEV  ;MORE UNITS
005670  103004                          BHIS    3$              ;NO-EXIT
005672  062737  000002  005704          ADD     #2,SETPTR       ;YES-GET TABLE ENTRY
005700  000756                          BR      1$              ;CONFIGURE THAT UNIT
005702  000207                  3$:     RETURN
005704  000000                  SETPTR: .WORD
```

```
                                        ;++
                                        ; CLRALL - CLEARS INPUT BUFFER FOR RESPONSE FROM UNIT.
                                        ;
                                        ; INPUTS:       BLKTBL - TOP OF DATA BLOCK ALLOCATION TABLE
                                        ;               LSTDEV - ADDR. OF LAST UNIT'S DATA BLOCK
                                        ;
                                        ; OUTPUTS:      ALL UNITS BUFFERS CLEARED.
                                        ;
                                        ; CALLS:        CLRBUF
                                        ;--
005706  012737  003346  006000  CLRALL:: MOV     #BLKTBL,CLRPTR  ;TOP OF TABLE OF ADDRESSES
005714  017705  000060          1$:     MOV     @CLRPTR,R5      ;GET DATA BLOCK
005720  004737  005746                  CALL    CLRBUF          ;CLEAR IT'S RECEIVE BUFFER
005724  023727  006000  003364          CMP     CLRPTR,#LSTDEV  ;LAST DEV?
005732  103004                          BHIS    2$              ;YES
005734  062737  000002  006000          ADD     #2,CLRPTR       ;-->NEXT
005742  000764                          BR      1$              ;CONTINUE
005744  000207                  2$:     RETURN
```

```
                        ;++
                        ; CLRBUF - CLEARS 1 UNIT'S INPUT BUFFER.
                        ; INPUTS:  RCVBUF(R5) IS BUFFER START
                        ;          RCBFSZ - SIZE OF RECEIVE BUFFER IN BYTES
                        ;          RCBFSZ IS SIZE OF BUFFER
                        ; OUTPUTS:  CLEARED AREA.
                        ;--

005746                  CLRBUF:: PUSH   R0              ;SAVE R0
005746  010046                                         MOV     R0,-(SP)

005750                          PUSH    R4              ;SAVE R4
005750  010446                                         MOV     R4,-(SP)

005752  016500  000102          MOV     RCVBUF(R5),R0   ;GET ADDRESS OF BUFFER
005756  012704  001036          MOV     #RCBFSZ,R4      ;SIZE IN BYTES
005762  005020          1$:     CLR     (R0)+           ;CLEAR IT
005764  162704  000002          SUB     #2,R4           ;2 BYTES LESS
005770  001374                  BNE     1$              ;MORE
005772                          POP     R4              ;RESTORE
005772  012604                                         MOV     (SP)+,R4

005774                          POP     R0              ;
005774  012600                                         MOV     (SP)+,R0

005776  000207                  RETURN                  ;EXIT
006000  000000          CLRPTR: .WORD
```

```
                                        ;++
                                        ; SETUP - CALLED WITHIN EACH TEST TO INSERT BEGINNING ADDRESS OF THE
                                        ;         TEST INTO ALL UNITS TEST PC'S.
                                        ; INPUTS:       TSTTOP LOADED WITH TEST ALGORITHMS STARTING ADDR.
                                        ;               BLKTBL - TOP OF DATA BLOCK ALLOCATION TABLE
                                        ;               LSTDEV - ADDR. OF LAST UNIT'S DATA BLOCK
                                        ;
                                        ; OUTPUTS:      TSTPC(R5) FOR ALL UNITS
                                        ;               DONE - CLEARED
                                        ;--

006002  005037  003320          SETUP:: CLR     DONE                    ;NOT DONE YET
006006  012737  003346  003322          MOV     #BLKTBL,IDPTR           ;TABLE TOP ADDR
006014  017705  175302          1$:     MOV     @IDPTR,R5               ;DEVICE'S DATA BLOCK
006020  013765  003324  000020          MOV     TSTTOP,TSTPC(R5)        ;INSERT PC FOR TOP OF TEST
006026  023727  003322  003364          CMP     IDPTR,#LSTDEV           ;ALL UNITS SET?
006034  103004                          BHIS    2$                      ;YES
006036  062737  000002  003322          ADD     #2,IDPTR                ;NO,GET NEXT POINTER
006044  000763                          BR      1$                      ;SET HIM UP
006046  000207                  2$:     RETURN                          ;DONE
```

```
                                ;++
                                ; RUN - IMPLEMENTS THE CALLS TO SEND PACKETS, RECEIVE PACKETS, THEN
                                ;       CHECK ANSWERS DURING TEST RUN TIME.
                                ; INPUTS:  DONE
                                ; OUTPUTS: NONE
                                ;--

006050  004737  006100          RUN::   CALL    NXTST           ;MAKE AND SEND NEXT PACK TO ALL
                                                                ;UNABORTED UNITS
006054  005737  003320                  TST     DONE            ;COMPLETE?
006060  001006                          BNE     2$              ;YES
006062  004737  007150                  CALL    GETANS          ;NO,GET ALL RESPONSES

006066                                  BREAK                   ;SUPERVISOR CHECK
006066  104422                                                          TRAP    C$BRK

006070  004737  010442                  CALL    CHKANS          ;CHECK ALL RESPONSES
006074  000765                          BR      RUN             ;CONTINUE TILL DONE
006076  000207                  2$:     RETURN
```

```
                                .SBTTL  NXTST / THE SCHEDULER

                                ;++
                                ; NXTST - DISPATCH EXECUTION USING EACH UN-ABORTED UNIT'S TEST PROGRAM
                                ;         COUNTER, (TSTPC(R5)). (THE POINTER TO THE TEST CODE THAT COMPRISES
                                ;         MAKING A PACKET AND SENDING IT. CHECKS FIRST FOR ANY UN-ABORTED UNIT
                                ;         THAT IS RETRYING EITHER A DATA ERROR OR A 'INDECIPHERABLE FLAG BYTE'
                                ;         ERROR, IN ORDER TO SERVICE ONLY THAT UNIT THIS PASS.  INITS
                                ;        NON-RETRYING UNITS IF NECESSARY. IF NO RETRIES,DISPATCH ALL
                                ;         UNITS IN ROUND ROBIN FASHION.
                                ;
                                ; INPUTS:       (IMPLIED) DATA BLOCKS.
                                ;               BLKTBL - TOP OF DATA BLOCK ALLOCATION TABLE
                                ;               LSTDEV - ADDR. OF LAST UNIT'S DATA BLOCK
                                ;
                                ; OUTPUTS:      ERRSF IF ALL UNITS ARE ABORTED.(TO NOTIFY APT)
                                ;               SYSTAT IS UPDATED
                                ;--

006100  000240          NXTST:: NOP             ;
006102  012737  03346  003310    MOV     #BLKTBL,DEVPTR  ;UNIT 0 TO START
006110  017705  75174    1$:    MOV     @DEVPTR,R5      ;GET DATA BLOCK
006114  005715                  TST     @R5             ;ABORTED?
006116  100504                  BMI     2$              ; YES... CHECK NEXT UNIT
006120  032715  000010    3$:    BIT     #BIT3,@R5       ;NO-RETRY 'BAD FLAG'?
006124  001040                  BNE     5$              ;YES...(SEND BREAK;THEN CMD PACK)
006126  032715  020000          BIT     #BIT13,@R5      ;NO-RETRYING STILL (NO END PACK YET)?
006132  001426                  BEQ     7$              ;NO...
006134  032715  000400          BIT     #BIT8,@R5       ;RETRYING A WRITE?
006140  001453                  BEQ     4$              ;NO...
006142                          SWAPIN                  ;YES-GET DEVICE REGESTERS
006142  016500  000006                                  MOV     6.(R5),R0
006146  016501  000010                                  MOV     8.(R5),R1
006152  016502  000012                                  MOV     10.(R5),R2
006156  016503  000014                                  MOV     12.(R5),R3
006162  016504  000016                                  MOV     14.(R5),R4

006166  020265  000206          CMP     R2,SAVCNT(R5)   ;CURRENT COUNT = SAVED COUNT? (WHERE WE STARTED)
006172  001036                  BNE     4$              ;NO...(CONTINUE SENDING DATA PACKS)
006174  042737  000004  003304  BIC     #BIT2,SYSTAT    ;YES-CLEAR RETRY FLAGS
006202  042715  020000          BIC     #BIT13,@R5      ;
006206  000450                  BR      2$              ;CHECK NEXT UNIT.
006210  032715  002000    7$:   BIT     #BIT10,@R5      ;NO-RETRY DATA ERROR?
006214  001445                  BEQ     2$              ;NO...ON TO NEXT UNIT
006216  052737  000002  003304  BIS     #BIT1,SYSTAT    ;SET RETRY STATUS TO 'DATA ERROR' TYPE
006224  000424                  BR      6$              ;YES...

006226                    5$:   SWAPIN                  ;GET DEVICE REGISTERS
006226  016500  000006                                  MOV     6.(R5),R0
006232  016501  000010                                  MOV     8.(R5),R1
006236  016502  000012                                  MOV     10.(R5),R2
006242  016503  000014                                  MOV     12.(R5),R3
006246  016504  000016                                  MOV     14.(R5),R4

006252  010265  000206          MOV     R2,SAVCNT(R5)   ;SAVE THE BYTE COUNT (FOR WRITE OPERATION)
                                                        ;TO MARK HOW MANY DATA PACKS TO SEND
006256  004737  013654          CALL    DOBRK           ;SEND INIT
```

```
1788 006262  032715  100000                    BIT     #BIT15,@R5      ;ABORTED?
1789 006266  001020                            BNE     2$              ;YES...
1790 006270  052737  000004  003304  4$:       BIS     #BIT2,SYSTAT    ;NOT ABORTED-SET RETRY STATUS
1791 006276                          6$:       SWAPIN                  ;GET DEVICE REGISTERS
     006276  016500  000006                                            MOV     6.(R5),R0
     006302  016501  000010                                            MOV     8.(R5),R1
     006306  016502  000012                                            MOV     10.(R5),R2
     006312  016503  000014                                            MOV     12.(R5),R3
     006316  016504  000016                                            MOV     14.(R5),R4

1792 006322  004775  000020                    JSR     PC,@TSTPC(R5)   ;DO TEST FOR
1793 006326  000477                            BR      NXTRET          ;THIS UNIT ONLY-EXIT
1794 006330  023727  003310  003364  2$:       CMP     DEVPTR,#LSTDEV  ;TRY NEXT UNIT?
1795 006336  103004                            BHIS    NXTST2          ;NO
1796 006340  062737  000002  003310            ADD     #2.,DEVPTR      ;YES,->NEXT
1797 006346  000660                            BR      1$              ;GET BLOCK
1798
1799 006350  005037  006530          NXTST2:   CLR     ABONM           ;HERE=NO RETRIES TO DO, NO UNIT ABORTED YET
1800 006354  012737  003346  003310            MOV     #BLKTBL,DEVPTR  ;-->UNIT 0 STORAGE BLOCK
1801 006362  017705  174722          PERDEV:   MOV     @DEVPTR,R5      ;R5-->NEXT DEVICE STORAGE BLOCK
1802
1803 006366  005715                  3$:       TST     @R5             ;ABORTED?
1804 006370  100426                            BMI     4$              ;YES
1805 006372  032715  040000                    BIT     #BIT14,@R5      ;SEND BREAK?
1806 006376  001407                            BEQ     6$              ;NO
1807 006400  004737  013654                    CALL    DOBRK           ;YES
1808 006404  032715  040000                    BIT     #BIT14,@R5      ;SUCCESSFUL INIT?
1809 006410  001016                            BNE     4$              ;NO ON TO NEXT UNIT
1810 006412  005715                            TST     @R5             ;ABORTED?
1811 006414  100414                            BMI     4$              ;YES-ON TO NEXT UNIT
1812 006416                          6$:       SWAPIN                  ;NO,GET DEVICE REGISTERS R0-R4 CONTAINING TEST PARAMETERS
     006416  016500  000006                                            MOV     6.(R5),R0
     006422  016501  000010                                            MOV     8.(R5),R1
     006426  016502  000012                                            MOV     10.(R5),R2
     006432  016503  000014                                            MOV     12.(R5),R3
     006436  016504  000016                                            MOV     14.(R5),R4

1813 006442  004775  000020                    JSR     PC,@TSTPC(R5)   ;INITIATE 1 PACKET TRANSMISSION AND RETURN
1814 006446  005715                  4$:       TST     @R5             ;ABORTED?
1815 006450  100002                            BPL     8$              ;NO-ON TO NEXT UNIT
1816 006452  005237  006530                    INC     ABONM           ;YES...ONE MORE TALLIED
1817 006456  023727  003310  003364  8$:       CMP     DEVPTR,#LSTDEV  ;ALL TU'S TRIED?
1818 006464  103004                            BHIS    5$              ;YES
1819 006466  062737  000002  003310            ADD     #2.,DEVPTR      ;NO THE ADDRESS+2=NEXT ADDRESS
1820 006474  000732                            BR      PERDEV          ;DO NEXT UNIT
1821 006476  022737  000010  006530  5$:       CMP     #8.,ABONM       ;ALL ABORTED?
1822 006504  001010                            BNE     NXTRET          ;NO
1823 006506                                    ERRSF   100.,NOMOR      ;YES!
     006506  104454                                                    TRAP    C$ERSF
     006510  000144                                                    .WORD   100
     006512  006532                                                    .WORD   NOMOR
     006514  000000                                                    .WORD   0
1824 006516                          11$:      BREAK                   ;SUPERVISOR BREAK
     006516  104422                                                    TRAP    C$BRK
1825 005520  005237  003336                    INC     ALLGON          ;SET DON'T-PRINT STATISTICS FLAG
1826 006524                                    DOCLN                   ;EXIT
     006524  104444                                                    TRAP    C$DCLN
```

```
 1827 006526  000207                 NXTRET: RETURN
 1828
 1829 006530  000000                 ABONM:  .WORD                    ;THE NUMBER OF ABORTED UNITS
 1830 006532     101     114     114 NOMOR:  .ASCIZ  /ALL UNITS ABORTED!/
 1831                                         .EVEN
```

```
                        .SBTTL  RSVP / XOFF AND SEND A PACKET TO ALL DEVICES

                        ;++
                        ; RSVP - SAVES TEST CODE PROGRAM COUNTER IN TSTPC(R5) AND UNIT'S REGIS-
                        ;        TERS.  IF NOT IN TEST 8, POINTS TO "XOFF" THAT PRECEEDS PACKET IN
                        ;        XMIT BUFFER AND SENDS PACKET WITH XOFF.  RETURNS TO SCHEDULER (NXTST)
                        ;        SO THAT OTHER UNITS PACKETS MAY BE FORMED, TO GET ALL UNITS WORKING
                        ;        AT ONCE.  IF IN TEST 8 AND THE UNIT IS NOT MODIFIED, SKIP REST OF
                        ;        ROUTINE.  IF IN TEST 8 AND THE UNIT IS MODIFIED DO NOT SEND XOFF AND
                        ;        PROCEED NORMALLY.
                        ; INPUTS:  (SP) CONTAINS UNITS PC TO SAVE SINCE RSVP WAS CALLED.  THE
                        ;          NUMBER PACKETS EXPECTED (XSPKNM), AND THE EXPECTED FLAGS AND
                        ;          BYTE COUNTS OF EACH (XSFLG, XSCNT...) ARE LOADED BY TEST CODE
                        ;          (MACROS).
                        ;          SNDCNT - # BYTES TO SEND
                        ;          REC(R5) - RECORD #
                        ;          TRBUF - BUFFER ADDR.
                        ;          XSPKNM(R5) - # EXPECTED
                        ;          RCVBUF(R5)
                        ;
                        ; OUTPUTS:  CMDSNT - UPDATED WITH PACKET OP CODE
                        ;           BLKER  - RECORD NUMBER STATISTICS UPDATED IF NOT RETRYING
                        ;                    AND COMMAND PACKET SENT.
                        ;           SUCCS(R5) - PRESET CLEAR
                        ;           STATUS WORD @R5 - BIT9 - DATA CHECK ERROR - CLEARED
                        ;                             BIT5 - "VERIFY" OPERATION
                        ;                             BIT4 - 0 = DATA PACK 1 = CMND
                        ;                             BIT8 - RD/WR OPERATION
                        ;           XSPTR - POINTS TO EXPECTED FLAG
                        ;           UPPER BYTE OF XSPKNM IS REPLICATED.
                        ;           PACKET POINTER (PKPTR(R5)) POINTS TO TOP OF UNITS RECEIVE BUFFER
                        ;           AREA (RCVBUF(R5)) FOR CURRENT UNIT.
                        ;--

006556  000240         RSVP::  NOP                     ;FINISH TEST
006560  012665  000020         MOV     (SP)+,TSTPC(R5) ;SAVE WHERE YOU WERE IN TEST BODY AND
006564                         SWAPOW                  ;SAVE TEST REGISTERS
006564  010065  000006                                 MOV     R0,6.(R5)
006570  010165  000010                                 MOV     R1,8.(R5)
006574  010265  000012                                 MOV     R2,10.(R5)
006600  010365  000014                                 MOV     R3,12.(R5)
006604  010465  000016                                 MOV     R4,14.(R5)

                                                       ;CORRECT FOR RETURN TO SCHEDULER
006610  022737  000002  003340         CMP     #2,TEST8        ;***** IS THIS TEST 8
006616  001007                         BNE     XFNSND          ;***** NO
006620  022765  000000  000210         CMP     #0,MRSP(R5)     ;***** IF SO, IS THIS UNIT MODIFIED
006626  001523                         BEQ     ENDRSP          ;***** YES
006630  012700  026174         NOXOFF: MOV     #TRBUF,R0       ;FOR NORMAL PACKET SEND
006634  000404                         BR      SND             ;SEND XOFF+PACKET
006636  012700  026173         XFNSND: MOV     #TRBUF-1,R0     ;POINT TO XOFF
006642  005265  000070                 INC     SNDCNT(R5)      ;ONE MORE TO SEND, TOO.
006646  004737  007100         SND:    CALL    SNDBYT          ;SEND BYTE
006652  005715                         TST     @R5             ;R5--> TO STATUS BLK
006654  100510                         BMI     6$              ;ABORTED? YES...QUIT
006656  005365  000070                 DEC     SNDCNT(R5)      ;NO, SEND MORE
```

```
1884 006662 001371                      BNE     SND             ;IF MORE TO SEND
1885 006664 012700 026174               MOV     #TRBUF,R0       ;-->BUFFER
1886 006670 016537 000064 003330        MOV     REC(R5),BLKER   ;PREPARE FOR RECEIVE
1887 006676 156565 000032 000033        BISB    XSPKNM(R5),XSPKNM+1(R5)  ;REPLICATE LO. BYTE TO HI FOR GTPAKS, CHKANS
1888 006704 005065 000076               CLR     SUCCS(R5)       ;NO SUCCESS YET
1889 006710 042715 001000               BIC     #BIT9,@R5       ;NO DATA CHK ERROR YET
1890 006714 016565 000102 000104        MOV     RCVBUF(R5),PKPTR(R5)  ;TOP OF RCV BUFFER GOES THE 1ST PACKET
1891 006722 012704 000034               MOV     #XSFLG,R4       ;FORM
1892 006726 060504                      ADD     R5,R4           ;ADDRESS
1893 006730 010465 000106               MOV     R4,XSPTR(R5)    ;OF 1ST XSFLG
1894
1895 006734 042715 000020               BIC     #BIT4,@R5       ;PRESET AS DATA PAK
1896 006740 121027 000002               CMPB    @R0,#RSCMND     ;WAS IT COMMAND PAK?
1897 006744 001054                      BNE     6$              ;NO...
1898 006746 116065 000002 000100        MOVB    2(R0),CMDSNT(R5)  ;YES-SAVE COMMAND
1899 006754 052715 000020               BIS     #BIT4,@R5       ;ITS CMND PAK
1900
1901 006760 032715 002000               BIT     #BIT10,@R5      ;RETRYING?
1902 006764 001044                      BNE     6$              ;YES-DON'T UPDATE ANY STATS OR CONDITION
1903 006766 126027 000002 000002        CMPB    2(R0),#RSSRD    ;NO,A READ?
1904 006774 001012                      BNE     4$              ;NO
1905 006776 042715 000400               BIC     #BIT8,@R5       ;(FOR HARD/SOFT LOGGING) RD/WR FLAG=0
1906 007002 004737 013504               CALL    WHCHDR          ;GET DRIVE
1907 007006 103403                      BCS     8$              ;
1908 007010 005265 000114               INC     RDN0(R5)        ;DRIVE 0
1909 007014 000402                      BR      4$              ;
1910 007016 005265 000116       8$:     INC     RDN1(R5)        ;DRIVE 1
1911
1912 007022 126027 000002 000003 4$:    CMPB    2(R0),#RSSWR    ;A WRITE?
1913 007030 001022                      BNE     6$              ;NO
1914 007032 052715 000400               BIS     #BIT8,@R5       ;YES, RD/WR FLAG=1
1915 007036 105760 000003               TSTB    3(R0)           ;VERIFY TOO?
1916 007042 001403                      BEQ     21$             ;NO
1917 007044 052715 000040               BIS     #BIT5,@R5       ;YES-SET VERIFY FLAG
1918 007050 000402                      BR      22$
1919 007052 042715 000040       21$:    BIC     #BIT5,@R5       ;(NO)-RESET VERIFY FLAG
1920 007056 004737 013504       22$:    CALL    WHCHDR          ;GET DRIVE NO
1921 007062 103403                      BCS     5$              ;CARRY=DR1
1922 007064 005265 000110               INC     WRTN0(R5)       ;# BLKS WRITTEN DR0
1923 007070 000402                      BR      6$              ;EXIT
1924
1925 007072 005265 000112       5$:     INC     WRTN1(R5)       ;# BLKS WRITTEN DRV1
1926 007076                     6$:
1927 007076 000207              ENDRSP: RETURN                  ;RETURN
```

```
                                .SBTTL  SNDBYT / OUTPUT A BYTE TO UNIT

                                ;++
                                ; SNDBYT - TEST 'READY' ON INTERFACE.  IF 'READY', SEND BYTE AND EXIT.
                                ;          IF TIMED OUT, LOG ERROR.
                                ; INPUTS - R0 = POINTER TO BUFFER
                                ;        - IMPLIED UNIT DATA BLOCK
                                ;        - CSNRDY - TIMEOUT CONSTANT
                                ; OUTPUTS - R0 IS INCREMENTED.
                                ; ERROR - NOT-READY-TO-SEND TIME OUT
                                ;--

007100                          SNDBYT:: PUSH   R1              ;ENTER R0-->BYTE
007100  010146                                                  MOV     R1,-(SP)


007102  013701  003342          4$:     MOV     CSNRDY,R1       ;GET TIMEOUT CONSTANT FOR NOT READY ERROR
007106  105775  000026          1$:     TSTB    @XMSR(R5)       ;READY TO SEND?
007112  100412                          BMI     2$              ;YES
007114                                  PUSH    R0              ;NO, SAVE R0
007114  010046                                                  MOV     R0,-(SP)


007116                                  BREAK                   ;MONITOR BREAK
007116  104422                                                                          TRAP    C$BRK
007120                                  POP     R0              ;RESTORE
007120  012600                                                  MOV     (SP)+,R0

                                ;
007122  005301                          DEC     R1              ;ABORTED?
007124  001370                          BNE     1$              ;NO
007126  012704  000056                  MOV     #TOSNDB,R4      ;YES,SET CODE FOR TIMEOUT ERROR
007132  004737  012500                  CALL    LOG             ;LOG IT
007136  000402                          BR      3$              ;QUIT
007140  112075  000030          2$:     MOVB    (R0)+,@XMDB(R5) ;SEND IT
007144                          3$:     POP     R1              ;RESTORE
007144  012601                                                  MOV     (SP)+,R1

007146  000207                          RETURN                  ;DONE
```

```
                                .SBTTL  GETANS / GETS RESPONSES ROUND ROBIN USING "XON"

                                ;++
                                ; GETANS - IF A UNIT IS RETRYING CLEAR HIS RECEIVE BUFFER (CLRBUF) AND GET
                                ;          HIS RESPONSE (GTPKS1), ELSE, CLEAR ALL BUFFERS (CLRALL) AND
                                ;          GET ALL RESPONSES (GTPKS8).
                                ; INPUTS:  SYSTAT - SYSTEM STATUS WORD.
                                ;
                                ; OUTPUTS: SERVST = -1 IF NO RETRIES.
                                ;--

007150  000240                  GETANS:: NOP                    ;1 UNIT IF RETRY;  ELSE ALL
007152  032737  000006  003304          BIT     #BIT1!BIT2,SYSTAT       ;RETRY?
007160  001010                        BNE     1$              ;YES
007162  012737  177777  010206        MOV     #-1,SERVST      ;PRESET NO UNITS SERVICED
007170  004737  005706                CALL    CLRALL          ;CLEAR ALL INPUT BUFFERS
007174  004737  007426                CALL    GTPKS8          ;GET ALL REPLYS
007200  000404                        BR      2$              ;EXIT
007202  004737  005746          1$:   CALL    CLRBUF          ;RETRY-CLEAR 1 UNIT ONLY
                                                              ;R5->UNIT BY NXTST
007206  004737  007216                CALL    GTPKS1          ;GET 1 REPLY
007212  000207                  2$:   RETURN                  ;DONE

007214  000000                  GETPTR: .WORD
```

```
                                .SBTTL  GTPKS1 / GET RETRY RESPONSE-1 UNIT

                                ;++
                                ; GTPKS1 - SENDS 'XON' TO UNIT, GETS FLAG BYTE (IF ANY), CHECKS IF IT IS
                                ;          WHAT WAS EXPECTED.  IF IT IS, USE EXPECTED BYTE COUNT(XSCNT).  IF
                                ;          NOT, CHECK IF PREMATURE-END PACK OR (SINCE MAINTENANCE MODE)
                                ;          IF IT'S A PREMATURE DATA PACK.  ADJUST COUNT, GET REST OF
                                ;          PACKET, AND REPEAT ABOVE UNTIL NO MORE PACKETS.
                                ; INPUTS:  (IMPLIED) UNITS DATA BLOCK
                                ;          RSNDSZ - END PACKET SIZE
                                ;
                                ; OUTPUTS: SYSTAT UPPER BYTE = FLAG BYTE RECEIVED
                                ;--

007216  000240          GTPKS1:: NOP                     ;R5->THE UNIT
007220  012703  000034          MOV     #XSFLG,R3        ;THE OFFSET VALUE OF FLAG
007224  060503                  ADD     R5,R3            ;FORM THE ABSOLUTE ADDRESS
007226  010301                  MOV     R3,R1            ;R3-->ADDR. OF EXPECTED FLAG
007230  062701  000002          ADD     #2.,R1           ;R1-->ADDR. OF EXPECTED COUNT
007234  012700  007424          MOV     #EXON,R0         ;R0=ADDRESS
007240  004737  007100          CALL    SNDBYT           ;XON THE DEVICE
                                                         ;*** TIME CRITICAL
007244  016500  000102          MOV     RCVBUF(R5),R0    ;***--> TO THE BUFFER
007250  116502  000033          MOVB    XSPKNM+1(R5),R2  ;***GET THE # OF PACKETS TO RECEIVE
007254  032702  177400          BIT     #177400,R2       ;***SIGN UN-EXTEND
007260  011137  003314  1$:     MOV     @R1,RCBCNT       ;***HOW MANY BYTES IT SHOULD BE
007264  011337  003312          MOV     @R3,RCFLG        ;***WHAT THE FIRST BYTE SHOULD BE
007270  004737  010212          CALL    GTBYTE           ;***GET THE ALL IMPORTANT FLAG
007274  032715  100000          BIT     #BIT15,@R5       ;TIMEOUT?
007300  001050                  BNE     4$               ;YES
007302  005300                  DEC     R0               ;-> BYTE RECIEVED
007304  111037  003305          MOVB    @R0,SYSTAT+1     ;SAVE IT AS FLAG BYTE
007310  121037  003312          CMPB    @R0,RCFLG        ;1ST BYTE WHAT WAS EXPECTED?
007314  001420                  BEQ     2$               ;YES
007316  121027  000002          CMPB    @R0,#RSEND       ;NO, WAS IT END PAK?
007322  001006                  BNE     14$              ;NO
007324  012737  000016  003314  MOV     #RSNDSZ,RCBCNT   ;YES, USE END SIZE FOR COUNT
007332  012702  000001          MOV     #1,R2            ;AND ASSUME IT'S LAST PACKET!
007336  000407                  BR      2$               ;CONTINUE RECEIVE
007340  121027  000001  14$:    CMPB    @R0,#RSDATA      ;WAS IT DATA?
007344  001026                  BNE     4$               ;NO,CHKANS MAY FIND INIT...
007346  012737  000204  003314  MOV     #RSDASZ,RCBCNT   ;YES, SET FOR DATA PAK SIZE
007354  005202                  INC     R2               ;ONE MORE PACK THAN EXPECTED (END PACK)

007356  005200          2$:     INC     R0               ;RESTORE TO -> NEXT BYTE
007360  005337  003314  5$:     DEC     RCBCNT           ;THAT'S ONE LESS BYTE TO GO
007364  001411                  BEQ     3$               ;DONE
007366  004737  010212          CALL    GTBYTE           ;GET REST OF PACKET
007372  005765  000074          TST     DLV(R5)          ;ERROR
007376  001011                  BNE     4$               ;YES-ALL OVER
007400  032715  100000          BIT     #BIT15,@R5       ;OR IF ABORTED
007404  001006                  BNE     4$               ;THEN QUIT
007406  000764                  BR      5$               ;CONTINUE RECEIVE

007410  005302          3$:     DEC     R2               ;ONE LESS PACKET TO GO
007412  001403                  BEQ     4$               ;MORE PACKETS IN TRANSACTION?
                                                         ;YES
```

```
 2043 007414  022121                    CMP     (R1)+,(R1)+     ;POINT TO NEW EXPECTED COUNT
 2044 007416  022323                    CMP     (R3)+,(R3)+     ;AND FLAG,
 2045 007420  000717                    BR      1$              ;AND RECEIVE,
 2046 007422  000207              4$:   RETURN                  ;RETURN
 2047
 2048 007424      020             EXON:  .BYTE   RSXON
 2049 007425      023             EXOFF: .BYTE   RSXOFF
```

```
                                .SBTTL  GTPKS8 / GET RESPONSES (NO RETRIES)

                                ;++
                                ; GTPKS8 - IF IN TEST 8 AND THE UNIT IS NOT MODIFIED, SKIP THE REST
                                ;          OF THE ROUTINE.  OTHERWISE;
                                ;          SET ALL ABORTED UNITS SERVICED (SERVST: BIT POSITION).  UNTIL
                                ;          ALL UNITS SERVICED (SERVST=0), IF NO MORE PACKETS, SET UNIT
                                ;          SERVICED, ELSE, GET A FLAG BYTE FROM UNIT, DECREMENTING THE
                                ;          NUMBER OF PACKETS LEFT.  CHECK TO SEE IF EXPECTED FLAG,
                                ;          ADJUST COUNT IF NOT, GET REST OF PACKET.  IF WAS DATA PACK,
                                ;          AND NOT IN TEST 8, SEND "XOFF" TO ENHANCE THROUGHPUT AND GO ON
                                ;          TO NEXT UNIT (IF ANY).  IF IN TEST 8, DO NOT SEND "XOFFF".
                                ; INPUTS:  (IMPLIED)UNITS DATA BLOCK POINTED TO BY R5.  NONE PASSED.
                                ;          RSNDSZ - END PACK SIZE
                                ;          RSDNSZ - DATA + END SIZE
                                ;
                                ; OUTPUTS: SYSTAT - UPPER BYTE=1ST BYTE RECEIVED, CURRENT UNIT
                                ;--

007426  000240                  GTPKS8:: NOP                    ;GET ALL UNITS RESPONSES XOFF IF DATA PAK (THROUGHPUT)
007430  022737  000002  003340          CMP     #2,TEST8        ;***** IS THIS TEST 8
007436  001006                        BNE     1$              ;***** NO
007440  022765  000000  000210        CMP     #0,MRSP(R5)     ;***** IF SO, IS THIS UNIT MODIFIED
007446  001002                        BNE     1$              ;***** YES, CONTINUE NORMALLY
007450  000137  010114                JMP     ENDGP8          ;***** ELSE, SKIP ROUTINE
007454  012737  003346  010210  1$:   MOV     #BLKTBL,GTPTR   ;->1ST
007462  017705  000522          GTAGIN: MOV     @GTPTR,R5       ;GET DATA BLOCK
007466  032715  100000                BIT     #BIT15,@R5      ;ABORTED?
007472  001403                        BEQ     2$              ;NO
007474  004737  010122                CALL    SETSRV          ;YES-SET' SERVICED' AND
007500  000564                        BR      GTDOWN          ;ON TO NEXT UNIT
007502  105765  000033          2$:   TSTB    XSPKNM+1(R5)    ;NO, ANY PACKETS LEFT?
007506  001003                        BNE     3$              ;YES
007510  004737  010122                CALL    SETSRV          ;NO-HE'S DONE
007514  000556                        BR      GTDOWN          ;SO ON TO NEXT UNIT
007516  105365  000033          3$:   DECB    XSPKNM+1(R5)    ;NOW ITS ONE LESS PACKET
007522  017537  000106  003312        MOV     @XSPTR(R5),RCFLG  ;GET EXPECTED FLAG
007530  062765  000002  000106        ADD     #2,XSPTR(R5)    ;--> COUNT
007536  017537  000106  003314        MOV     @XSPTR(R5),RCBCNT  ;AND EXPECTED COUNT
007544  022737  000002  003340        CMP     #2,TEST8        ;***** IF TEST 8
007552  001404                        BEQ     1$              ;***** DO NOT SEND XON
007554  012700  007424                MOV     #EXON,R0        ;-> XON
                                                              ;***TIME CRITICAL
007560  004737  007100                CALL    SNDBYT          ;***SEND IT
007564  016500  000104          1$:   MOV     PKPTR(R5),R0    ;***->WHERE 1ST BYTE GOES
007570  004737  010212                CALL    GTBYTE          ;***GET IT
007574  032715  100000                BIT     #BIT15,@R5      ;ABORTED?
007600  001403                        BEQ     4$              ;NO-CONTINUE
007602  105065  000033                CLRB    XSPKNM+1(R5)    ;YES-NO MORE PACKETS EXPECTED
007606  000521                        BR      GTDOWN          ;ON TO NEXT
007610  005300                  4$:   DEC     R0              ;-->BYTE JUST RECEIVED
007612  111037  003305                MOVB    @R0,SYSTAT+1    ;SAVE IT
007616  121037  003312                CMPB    @R0,RCFLG       ;IS IT WHAT EXPECTED?
007622  001436                        BEQ     GTOK            ;YES
007624  105065  000033          UNXPCT: CLRB  XSPKNM+1(R5)    ;NO, MUST BE LAST REPLY
007630  121027  000002                CMPB    @R0,#RSEND      ;MAYBE AN END PAK?
007634  001004                        BNE     4$              ;NO
```

```
                                                MOV     #RSNDSZ,RCBCNT  ;YES, USE PROPER COUNT
                                                BR      GTUM            ;AND GET IT
                                        4$:     CMPB    @R0,#RSDATA     ;IS IT DATA?
                                                BNE     GTDOWN          ;NO, ALL OVER, CHKANS WILL INIT UNIT
                                                MOV     #RSDNSZ,RCBCNT  ;YES, USE COUNT OF DATA + END PAK SURE TO FOLLOW
                                        GTUM:   INC     R0              ;WHERE TO STUFF THE REST
                                        5$:     DEC     RCBCNT          ;ONE DOWN
                                                BEQ     GTDOWN          ;NONE TO GO
                                                CALL    GTBYTE          ;MORE TO GO
                                                BIT     #BIT15,@R5      ;TIMEOUT?
                                                BNE     GTDOWN          ;YES
                                                TST     DLV(R5)         ;BUT DLV ERROR?
                                                BEQ     5$              ;NO
                                                CLRB    XSPKNM+1(R5)    ;YES-LAST TIME
                                                BR      GTDOWN          ;ON TO NEXT

                                        GTOK:   INC     R0              ;NEXT PLACE IN BUFFER
                                        1$:     CMP     #2,TEST8        ;***** IF NOT TEST 8
                                                BNE     7$              ;***** DO NOT SEND 'CONT'
                                                PUSH    R0              ;*****SEND 'CONT' FOR MRSP
007732  010046                                                          MOV     R0,-(SP)


                                                MOV     #MODRSP,R0      ;
                                                CALL    SNDBYT          ;
                                                NOP
                                                MOV     #1,MRSDLY       ;***** ANOTHER DELAY
                                        2$:     CLR     R0              ;*****
                                        3$:     DEC     R0              ;***** TO GET AN ERROR
                                                BNE     3$              ;***** IF MRSP DOESN'T WORK
                                                DEC     MRSDLY          ;*****
                                                BNE     2$              ;
                                                POP     R0              ;
007770  012600                                                          MOV     (SP)+,R0

                                        7$:     DEC     RCBCNT          ;MORE BYTES?
                                                BEQ     4$              ;NO-ALL DONE
                                                CALL    GTBYTE          ;YES-GET IT
                                                BIT     #BIT15,@R5      ;TIMEOUT?
                                                BNE     GTDOWN          ;YES
                                                TST     DLV(R5)         ;ERROR?
                                                BEQ     1$              ;NO
                                                CLRB    XSPKNM+1(R5)    ;LAST TIME
                                                BR      GTDOWN          ;EXIT
                                        4$:     CMPB    #RSDATA,@PKPTR(R5)  ;WAS DATA?
                                                BNE     GTDOWN          ;NO, ALL DONE
                                                MOV     R0,PKPTR(R5)    ;START OF NEXT PACK NEXT TIME
                                                MOV     #EXOFF,R0       ;XOFF AND SEND TO
                                                CALL    SNDBYT          ;ENHANCE THROUGHPUT
                                        GTDOWN: ADD     #2.,XSPTR(R5)   ;NEXT XSFLG FOR NEXT TRY
                                                CMP     GTPTR,#LSTDEV   ;DONE ONE CYCLE ALL UNITS?
                                                BHIS    1$              ;YES
                                                ADD     #2,GTPTR        ;NEXT UNIT
                                                JMP     GTAGIN          ;CONTINUE RECEIVE
                                        1$:     TSTB    SERVST          ;DONE SERVICING ALL PAKS
                                                                        ;FROM ALL UNITS?
                                                BEQ     ENDGP8          ;YES
```

```
   2161 010110  000137  007426                    JMP     GTPKS8          ;NO, KEEP TRYING
   2162 010114  000207                ENDGP8: RETURN                      ;RETURN
   2163
   2164 010116     020                MODRSP: .BYTE   RSCONT
   2165                                       .EVEN
   2166 010120  000000                MRSDLY: .WORD
```

```
                                .SBTTL  SETSRV / SET UNIT SERVICED

                                ;++
                                ; SETSRV - RESET THE BIT IN 'SERVST' CORRESPONDING TO THE UNIT NUMBER.
                                ; INPUTS - SERVST - 'SERVICED' WORD
                                ;        - @R5 = UNIT # (BITS 0, 1, 2)
                                ; OUTPUTS - SERVST MODIFIED
                                ;--

010122                          SETSRV: PUSH    R5              ;SET UNIT SERVICED
010122  010546                          MOV     R5,-(SP)

010124                                  PUSH    R0
010124  010046                          MOV     R0,-(SP)

010126  011505                          MOV     @R5,R5          ;GET STAT WD
010130  042705  177770                  BIC     #177770,R5      ;MASK UNIT #
010134  012700  010166                  MOV     #SRVTBL,R0      ;->TOP OF BIT TABLE
010140  005705                  1$:     TST     R5              ;RIGHT ONE?
010142  001404                          BEQ     2$              ;YES
010144  062700  000002                  ADD     #2,R0           ;NO, ->NEXT
010150  005305                          DEC     R5              ;1 LESS
010152  000772                          BR      1$              ;CONTINUE
010154  041037  010206          2$:     BIC     @R0,SERVST      ;MOW IT DOWN
010160                                  POP     R0
010160  012600                          MOV     (SP)+,R0

010162                                  POP     R5
010162  012605                          MOV     (SP)+,R5

010164  000207                          RETURN                  ;RETURN

010166  000001                  SRVTBL: .WORD   BIT0            ;BIT POSITION LOOKUP TABLE
010170  000002                          .WORD   BIT1
010172  000004                          .WORD   BIT2
010174  000010                          .WORD   BIT3
010176  000020                          .WORD   BIT4
010200  000040                          .WORD   BIT5
010202  000100                          .WORD   BIT6
010204  000200                          .WORD   BIT7

010206  000000                  SERVST: .WORD
010210  000000                  GTPTR:  .WORD
```

```
                                .SBTTL  GTBYTE / GET A BYTE FROM UNIT

                                ;++
                                ; GTBYTE - TEST INTERFACE FOR 'READY-TO-RECEIVE' AND INPUT A BYTE, IF
                                ;          SO.  IF NOT, THE FOLLOWING OCCURS:  SEND 'XOFF' TO UNIT IN
                                ;          PREPARATION FOR ^C CHECK ('BREAK' TO SUPERVISOR).  WAIT
                                ;          TO SEE IF A CHARACTER SLOPS OVER DUE TO UART LATENCY. IF
                                ;          ONE DOES THEN MIGHT AS WELL GET IT AND SEND 'XON' TO GET
                                ;          THE REST OF THE MESSAGE, OTHERWISE, 'BREAK'.  THEN SEND
                                ;          'XON', AND TEST FOR LONG TIMEOUT (A 30 SECOND REWIND).  IF SO,
                                ;          LOG ERROR, OTHERWISE REPEAT THE ABOVE UNTIL READY OR TIME OUT.
                                ;          REMEMBER TO PRESERVE R0 SINCE THE 'BREAK' TRAP CLOBBERS IT.
                                ;
                                ; INPUTS  - R0 POINTS TO INPUT BUFFER
                                ;         - IMPLIED UNITS DATA BLOCK
                                ;         - CSRCVB TIME OUT MULTIPLIER
                                ;
                                ; OUTPUTS - R0 IS INCREMENTED
                                ;         - DLV (R5) NON-ZERO ON INTERFACE ERROR.
                                ;
                                ; ERROR   - TIME OUT ON RECEIVE
                                ;--

010212  005037  010436          GTBYTE:: CLR    GBTMP           ;TIMEOUT REGISTER
010216  013704  003344                   MOV    CSRCVB,R4       ;TIMEOUT ERROR CONSTANT (MULTIPLIER)
010222  105775  000022          1$:      TSTB   @RCSR(R5)       ;READY?
010226  100013                           BPL    3$              ;NO
010230  017565  000024  000074         MOV    @RCDB(R5),DLV(R5)  ;GET ERROR + BYTE
010236  116520  000074                 MOVB   DLV(R5),(R0)+   ;COPY BYTE TO BUFFER
010242  005765  000074                 TST    DLV(R5)         ;ERROR?
010246  100472                         BMI    4$              ;YES-EXIT
010250  005065  000074                 CLR    DLV(R5)         ;NO-RESET
010254  000467                         BR     4$              ;AND EXIT
010256  005337  010436          3$:      DEC    GBTMP           ;DEC T.O. CONSTANT
010262  001357                         BNE    1$              ;STILL VALID

                                ;CODE TO SEE ^C DURING LONG SEEK OR REWIND
                                                               ;HERE GBTMP=0
010264  010037  010440                 MOV    R0,GBTMP2       ;R0 MUST BE PRESERVED!
010270  012700  007425                 MOV    #EXOFF,R0       ;QUIET THE DEVICE
010274  004737  007100                 CALL   SNDBYT          ;BY SENDING XOFF
010300  105775  000022          6$:      TSTB   @RCSR(R5)       ;CHARACTER SLOP OVER?
010304  100415                         BMI    5$              ;YES
010306  005337  010436                 DEC    GBTMP           ;NO-WAIT A WHILE
010312  105737  010436                 TSTB   GBTMP           ;DONE WAITING?
010316  001370                         BNE    6$              ;NO
010320                                 BREAK                  ;YES-NO SLOP OVER
        010320  104422                                                  TRAP    C$BRK
010322  012700  007424                 MOV    #EXON,R0        ;START DEVICE TALKING
010326  004737  007100                 CALL   SNDBYT          ;AGAIN
010332  013700  010440                 MOV    GBTMP2,R0       ;RESTORE R0
010336  000426                         BR     7$              ;END KLUGE
010340  013700  010440          5$:      MOV    GBTMP2,R0       ;RESTORE R0
010344  017565  000024  000074         MOV    @RCDB(R5),DLV(R5)  ;GET ERROR + BYTE
010352  116520  000074                 MOVB   DLV(R5),(R0)+   ;COPY BYTE TO BUFFER
010356  005765  000074                 TST    DLV(R5)         ;ERROR?
010362  100403                         BMI    17$             ;YES-EXIT
```

```
2262 010364  005065  000074                CLR     DLV(R5)          ;NO-CLEAR
2263 010370  000400                        BR      17$              ;EXIT
2264 010372  010037  010440        17$:    MOV     R0,GBTMP2        ;AGAIN SAVE R0
2265 010376  012700  007424                MOV     #EXON,R0         ;RESTORE TO TALKING STATE
2266 010402  004737  007100                CALL    SNDBYT           ;BY SENDING 'XON'
2267 010406  013700  010440                MOV     GBTMP2,R0        ;RESTORE R0
2268 010412  000410                        BR      4$               ;DONE
2269 010414  005037  010436        7$:     CLR      GBTMP
2270 010420  005304                        DEC     R4               ;TIMEOUT?
2271 010422  001277                        BNE     1$               ;NO
2272 010424  012704  000050                MOV     #TORCVB,R4       ;YES
2273 010430  004737  012500                CALL    LOG              ;LOG ERROR.
2274 010434  000207                4$:     RETURN                   ;RETURN
2275 010436  000000                GBTMP:  .WORD   0
2276 010440  000000                GBTMP2: .WORD   0
```

```
                                .SBTTL  CHKANS / CHECK DEVICE(S) RESPONSE

                                ;++
                                ; CHKANS - AS IN "GETANS", IF RETRYING DO ONLY 1 UNIT ELSE DO ALL NON-
                                ;          ABORTED UNITS.  NOTE, IF IN TEST 8 AND THE UNIT IS NOT
                                ;          MODIFIED DO NOT CHECK UNIT.
                                ; INPUTS:  IMPLIED SYSTAT BIT1 (RETRYING)
                                ;          BLKTBL - TOP OF DATA BLOCK ALLOCATION TABLE
                                ;          LSTDEV - ADDR. OF LAST UNIT'S DATA BLOCK
                                ;
                                ; OUTPUTS: NONE PASSED.
                                ;--

010442  000240                  CHKANS:: NOP                    ;IF RETRY THEN CHECK ONE
                                                                ;ELSE CHECK ALL
010444  032737  000006  003304          BIT     #BIT1!BIT2,SYSTAT       ;RETRYING?
010452  001403                        BEQ     CHK8            ;NO DO NORMAL
010454  004737  010552                CALL    CHKPKS          ;YES DO SINGLE UNIT
                                                                ;R5 -> UNIT
010460  000432                        BR      CHKANR          ;ALL DONE

010462  012737  003346  010550  CHK8:   MOV     #BLKTBL,CHKPTR  ;YOU KNOW ... TOP OF TABLE
010470  017705  000054          2$:     MOV     @CHKPTR,R5      ;GET UNIT'S BLOCK ADDRESS
010474  032715  100000                  BIT     #BIT15,@R5      ;ABORTED?
010500  001012                        BNE     3$              ;YES
010502  022737  000002  003340        CMP     #2,TEST8        ;***** IS THIS TEST 8
010510  001004                        BNE     1$              ;***** NO-CONTINUE NORMALLY
010512  022765  000000  000210        CMP     #0,MRSP(R5)     ;***** IF SO, IS THIS UNIT MODIFIED
010520  001402                        BEQ     3$              ;***** NO SKIP NEXT INSTR
010522  004737  010552          1$:     CALL    CHKPKS          ;NO, DO THIS GUY
010526  023727  010550  003364  3$:     CMP     CHKPTR,#LSTDEV  ;ALL DONE?
010534  103004                        BHIS    CHKANR          ;YES
010536  062737  000002  010550        ADD     #2,CHKPTR       ;NO,-->NEXT DEVICE
010544  000751                        BR      2$              ;DO DA

010546  000207                  CHKANR: RETURN

010550  000000                  CHKPTR: .WORD
```

```
                                .SBTTL  CHKPKS / DECIPHERS RESPONSE OF UNIT POINTED TO BY R5 /

                                ;++
                                ; CHKPKS - FOR UNIT R5 AND FOR ALL PACKETS, CHECK TO SEE IF PACKET IS DATA OR
                                ;          END PACK, CHECK CHECKSUMS, COMPARE DATA IF DATA PACK, CHECK
                                ;          SUCCESS CODE IF END.  IF UNKNOWN PACKET TYPE, CHECK FOR INTERFACE
                                ;          ERROR.  IF "CONTINUE" FALL THROUGH.  IF "INIT" SET "SEND
                                ;          BREAK" FLAG.  CALL "LOG" WITH R4=ERROR NUMBER IF ERROR.
                                ;          THIS ROUTINE IS ALSO USED TO DETERMINE THE PROTOCOL OF A UNIT.  IN
                                ;          THE FIRST PART OF TEST 8 A GET CHARACTERISTICS COMMAND PACKET WAS
                                ;          SENT TO THE TU58.  IF THE RESPONSE WAS A DATA PACKET, WHICH IS
                                ;          EXPECTED, THEN THE UNIT IS NOT MODIFIED, AND THE MRSP FLAG IS
                                ;          CLEARED.  IF THE RESPONSE IS AN END PACKET, WHICH WOULD BE
                                ;          HANDLED BY THIS ROUTINE AS AN UNKNOWN, THEN THE UNIT IS MODIFIED,
                                ;          AND THE MRSP FLAG IS SET.
                                ; INPUTS:  (IMPLIED) UNITS DATA BLOCK
                                ;
                                ; OUTPUTS: ERRORS - DLV ERROR
                                ;                 - UNKNOWN FLAG BYTE ERROR
                                ;                 - CHECKSUM ERROR
                                ;                 - DATA COMPARE ERROR
                                ;          R4 = ERROR NUMBER
                                ;          SYSTAT UPPER BYTE = 1ST BYTE OF RESPONSE
                                ;--

010552  000240                  CHKPKS:: NOP                    ;CHECK WHAT WAS RECIEVED
010554  042715  000010                  BIC     #BIT3,@R5       ;CLEAR 'BAD FLAG' RETRY BIT
010560  016500  000102                  MOV     RCVBUF(R5),R0   ;GET BUFFER ADDR.
010564  116502  000032                  MOVB    XSPKNM(R5),R2   ;AND # OF PACKETS EXPECTED
010570  012703  000034                  MOV     #XSFLG,R3       ;THE OFFSET VALUE
010574  060503                          ADD     R5,R3           ;R3-->THIS UNIT XSFLG AGAIN
010576  010301                          MOV     R3,R1           ;COPY TO R1
010600  062701  000002                  ADD     #2,R1           ;R1-->XSBCNT FOR 1ST PACKET
010604  010065  000104          1$:     MOV     R0,PKPTR(R5)    ;POINT TO PACKET
010610  111037  003305                  MOVB    @R0,SYSTAT+1    ;SAVE RCV'D BYTE
010614  011137  003314                  MOV     @R1,RCBCNT      ;GET COUNT
010620  011337  003312                  MOV     @R3,RCFLG       ;AND FLAG
010624  121013                          CMPB    @R0,@R3         ;1ST BYTE=EXPECTED?
010626  001057                          BNE     5$              ;UH OH...
010630  121027  000020                  CMPB    @R0,#RSCONT     ;OK, IS IT 1 BYTE?
010634  001534                          BEQ     7$              ;YES...ONTO NEXT PACK
                                                                ;NO, SO > 1 BYTE (NEVER EXPECT INIT!)
010636  013704  003314                  MOV     RCBCNT,R4       ;EXPECTED, SO COUNT MUST BE RIGHT
010642  005744                          TST     -(R4)           ;ADJUST FROM RECEIVE COUNT TO COUNT FOR CHECKSUM
010644  004737  013614                  CALL    CKCKSM          ;CHECK CHECKSUM
010650  103005                          BCC     2$              ;NO CARRY...NO INCORRECT
010652  012704  000022                  MOV     #BDCHK,R4       ;ERROR
010656  004737  012500                  CALL    LOG             ;LOG IT
010662  000521                          BR      7$              ;ON TO NEXT PACK
010664  122710  000002          2$:     CMPB    #RSEND,(R0)     ;END PAK?
010670  001005                          BNE     3$              ;NO
010672  004737  011146                  CALL    CHKEND          ;YES-CHECK
010676  012702  000001                  MOV     #1,R2           ;LAST PACKET
010702  000511                          BR      7$              ;AND FALL THROUGH
010704  122710  000001          3$:     CMPB    #RSDATA,@R0     ;DATA PAK?
010710  001012                          BNE     4$              ;NO
010712  022737  000001  003340          CMP     #1,TEST8        ;***** IS THIS TEST 8
```

```
2376 010720  001003                    BNE     11$             ;***** NO-CONTTINUE NORMALLY
2377 010722  005065  000210            CLR     MRSP(R5)        ;***** CLR MRSP FLAG
2378 010726  000402                    BR      12$             ;***** SKIP INSTR
2379 010730  004737  014414  11$:      CALL    COMPAR          ;YES-CHECK DATA
2380 010734  000474          12$:      BR      7$              ;ALL DONE?
2381 010736  052715  020010  4$:       BIS     #BIT3!BIT13,@R5 ;SET 'BAD FLAG' RETRY FLAGS
2382 010742  012704  000010            MOV     #OTL,R4         ;OUT TO LUNCH
2383 010746  005765  000074            TST     DLV(R5)         ;AH,BUT DLV ERROR?
2384 010752  001402                    BEQ     20$             ;NO
2385 010754  012704  000012            MOV     #OVRN,R4        ;YES-USE CORRECT ERROR #
2386 010760  004737  012500  20$:      CALL    LOG             ;TALLY
2387 010764  000467                    BR      8$              ;DONE
2388
2389                           ;HERE CHECKS UNEXPECTED RESPONSE
2390
2391 010766  122710  000004  5$:       CMPB    #RSINIT,@R0     ;INIT?
2392 010772  001007                    BNE     6$              ;NO
2393 010774  052715  020010            BIS     #BIT3!BIT13,@R5 ;YES-SET RETRY FLAGS
2394 011000  012704  000006            MOV     #RCINIT,R4      ; WE GOT AN INIT
2395 011004  004737  012500            CALL    LOG             ;TALLY IT
2396 011010  000455                    BR      8$              ;DONE
2397 011012  122710  000001  6$:       CMPB    #RSDATA,@R0     ;DATA PAK?
2398 011016  001013                    BNE     9$              ;NO
2399 011020  012704  000204            MOV     #RSDASZ,R4      ;YES, USE DATA SIZE
2400 011024  005744                    TST     -(R4)           ;ADJUST FOR CHKSUM
2401 011026  004737  013614            CALL    CKCKSM          ;AND CHECK
2402 011032  103430                    BCS     10$             ;GOOF
2403 011034  004737  014414            CALL    COMPAR          ;OK, HOW'S THE DATA?
2404                                                           ;EXPECTED END, GOT
2405                                                           ;DATA + END.
2406 011040  062700  000204            ADD     #RSDASZ,R0      ;POINT TO END PACK
2407 011044  000657                    BR      1$              ;CHECK IT, USE SAME XSFLG
2408
2409 011046  122710  000002  9$:       CMPB    #RSEND,(R0)     ;END?
2410 011052  001331                    BNE     4$              ;NO-OUT TO LUNCH
2411 011054  012704  000016            MOV     #RSSNSZ,R4      ;YES, TOTAL SIZE MINUS
2412 011060  005744                    TST     -(R4)           ;TWO (THE CHKSUM)
2413 011062  004737  013614            CALL    CKCKSM          ;CHECK IT
2414 011066  103412                    BCS     10$             ;OOPS
2415 011070  022737  000001  003340    CMP     #1,TEST8        ;***** IS THIS TEST 8
2416 011076  001003                    BNE     13$             ;***** NO-CONTINUE NORMALLY
2417 011100  012765  000001  000210    MOV     #1,MRSP(R5)     ;***** IF SO, SET THE MRSP FLAG
2418 011106  004737  011146  13$:      CALL    CHKEND          ;OK,NOW TEST SUC. CODE
2419
2420 011112  000414                    BR      8$              ;ALL DONE
2421
2422 011114  012704  000022  10$:      MOV     #BDCHK,R4       ;CHECKSUM ERROR
2423 011120  004737  012500            CALL    LOG
2424 011124  000407                    BR      8$              ;EXIT
2425
2426 011126  005302          7$:       DEC     R2              ;ANY PACKETS LEFT TO CHECK?
2427 011130  001405                    BEQ     8$              ;NO, ALL DONE
2428 011132  063700  003314            ADD     RCBCNT,R0       ;YES, POINT TO NEXT PACKET
2429 011136  022121                    CMP     (R1)+,(R1)+     ;POINT TO NEXT EXPECTED COUNT
2430 011140  022323                    CMP     (R3)+,(R3)+     ;AND EXPECTED FLAG
2431 011142  000620                    BR      1$              ;TRY ANOTHER,THEY'RE SMALL
2432 011144  000207          8$:       RETURN                  ;RETURN
```

```
                                .SBTTL  CHKEND / CHECK SUCCESS AND DETERMINE RETRY STATUS /

                                ;++
                                ; CHKEND - IF RETRYING;DETERMINE IF DATA ERROR OR BAD FLAG BYTE ERROR RETRY.
                                ;
                                ;       IF RETRYING BAD FLAG:RESET RETRY FLAG(SINCE OPERATION IS COMPLETE),
                                ;       AND CHECK SUCCESS CODE.
                                ;        IF RETRYING DATA ERROR ; CHECK SUCCESS CODE AND IF 0, PRINT RECOVERED, LOG
                                ;          SOFT ERROR, END RETRY STATUS.  IF NOT 0 AND WAS STILL 'DATA
                                ;          CHECK'' ERROR - DETERMINE WHETHER TO CONTINUE ANOTHER RETRY OR
                                ;          LOG 'UNRECOVERABLE'' ERROR.
                                ;
                                ;          IF NOT RETRYING DATA ERROR;  CHECK IF 'DATA CHECK' ERROR SUCCESS CODE,
                                ;          AND IF SO,START RETRY, ELSE EXIT.
                                ; INPUTS:  IMPLIED  UNITS DATA BLOCK
                                ; OUTPUTS: RETRY (SYSTAT BIT 1AND 2), (BIT10 @R5) RESET IF RETRYING.
                                ;          -  DATA COMARE ERROR (BIT6 @R5) CLEARED.
                                ;          -   REDUCED/NORMAL GAIN (BIT7 @R5) ADJUSTED
                                ;--

011146  000240                  CHKEND:: NOP
011150                                   PUSH    R0              ;R0 --> END PAK
011150  010046                                   MOV     R0,-(SP)

011152                                   PUSH    R4
011152  010446                                   MOV     R4,-(SP)

011154  032737  000006  003304  1$:      BIT     #BIT1.BIT2,SYSTAT       ;RETRYING?
011162  001406                         BEQ     NOREE           ;NO-CHECK NORMALLY
011164  032737  000004  003304         BIT     #BIT2,SYSTAT    ;IS IT BAD FLAG TYPE?
011172  001454                         BEQ     CHKREE          ;NO(DATA TYPE)
011174  042715  020000                 BIC     #BIT13,@R5      ;YES, SO IF END PACK THEN RETRY'S COMPLETE
011200  004737  012164          NOREE:  CALL    CHKSUC          ;CHECK SUCCESS CODE
011204  032715  100000                 BIT     #BIT15,@R5      ;ABORTED?
011210  001402                         BEQ     3$              ;NO,CONTINUE
011212  000137  011670                 JMP     CHKRET          ;YES,EXIT
011216  105765  000077          3$:     TSTB    SUCCS+1(R5)     ;NO; HOW'D WE DO?
011222  001013                         BNE     CHKERR          ;NOT SO GOOD.
011224  032715  000100                 BIT     #BIT6,@R5       ;OK, HOST FIND DATA PAK ERROR?
011230  001002                         BNE     2$              ;YES
011232  000137  011670                 JMP     CHKRET          ;NO
011236  012704  000014          2$:     MOV     #BDCOM,R4       ;YES;  JUST BAD DATA-NO DATACHK ERR
011242  004737  012500                 CALL    LOG             ;BAD DATA IN PACKET
011246  000137  011670                 JMP     CHKRET          ;QUIT
011252  032715  001000          CHKERR: BIT     #BIT9,@R5       ;BAD SUCCESS; TU DATA CHK ERROR?
011256  001002                         BNE     1$              ;YES
011260  000137  011670                 JMP     CHKRET          ;NO. ALL DONE.
011264  052715  002000          1$:     BIS     #BIT10,@R5      ;YES-START RETRY
011270  012765  000001  000002         MOV     #1,RETRY(R5)    ;CALL IT 1ST
011276                                   PRINTX  #RTRYN,RETRY(R5)  ;** PRINT **
011276  016546  000002                                                   MOV     RETRY(R5),-(SP)
011302  012746  012050                                                   MOV     #RTRYN,-(SP)
011306  012746  000002                                                   MOV     #2,-(SP)
011312  010600                                                           MOV     SP,R0
011314  104415                                                           TRAP    C$PNTX
```

```
        011316  062706  000006                                                          ADD     #6,SP
 2481   011322  000562                  BR      CHKRET          ;ALL DONE
 2482   011324  004737  012164  CHKREE: CALL    CHKSUC          ;CHECK SUCCESS CODE
 2483   011330  105765  000077          TSTB    SUCCS+1(R5)     ; SUCCESSFUL YET?
 2484   011334  001054                  BNE     UNSUC           ;NO, CHECK COUNT
 2485   011336                          PRINTX  #RECOV,RETRY(R5)
        011336  016546  000002                                                          MOV     RETRY(R5),-(SP)
        011342  012746  011710                                                          MOV     #RECOV,-(SP)
        011346  012746  000002                                                          MOV     #2,-(SP)
        011352  010600                                                                  MOV     SP,R0
        011354  104415                                                                  TRAP    C$PNTX
        011356  062706  000006                                                          ADD     #6,SP
 2486   011362  105715                  TSTB    (R5)            ;DETERMINE THRESHOLD
 2487   011364  100411                  BMI     2$              ;IT'S MODIFIED
 2488   011366                          PRINTX  #THRSLO         ;NORMAL
        011366  012746  011770                                                          MOV     #THRSLO,-(SP)
        011372  012746  000001                                                          MOV     #1,-(SP)
        011376  010600                                                                  MOV     SP,R0
        011400  104415                                                                  TRAP    C$PNTX
        011402  062706  000004                                                          ADD     #4,SP
 2489   011406  000410                  BR      3$
 2490   011410                  2$:     PRINTX  #THRSHI         ;ENHANCED
        011410  012746  012016                                                          MOV     #THRSHI,-(SP)
        011414  012746  000001                                                          MOV     #1,-(SP)
        011420  010600                                                                  MOV     SP,R0
        011422  104415                                                                  TRAP    C$PNTX
        011424  062706  000004                                                          ADD     #4,SP
 2491   011430  032715  000400          3$:     BIT     #BIT8,@R5       ;WRITE OR READ OPERATION?
 2492   011434  001003                  BNE     4$              ;WRITE
 2493   011436  012704  000002          MOV     #SFTRD,R4       ;READ
 2494   011442  000402                  BR      5$
 2495   011444  012704  000004          4$:     MOV     #SFTWR,R4       ;WRITE
 2496   011450  004737  012500          5$:     CALL    LOG
 2497   011454  005065  000002          CLR     RETRY(R5)       ;RESTORE TO NORMAL STATE
 2498   011460  042715  002200          BIC     #BIT10!BIT7,@R5 ;NO RETRY, NORM THRESHOLD
 2499   011464  000501                  BR      CHKRET          ;QUIT
 2500
 2501   011466  000240          UNSUC:  NOP                     ;RETRYING;  SEE IF HARD YET
 2502   011470  032715  001000          BIT     #BIT9,@R5       ;TU DATA CHECK ERROR?
 2503   011474  001015                  BNE     2$              ;YES
 2504   011476                          PRINTB  #RETERR         ;NO-''OTHER-ERROR'' ERROR
        011476  012746  012112                                                          MOV     #RETERR,-(SP)
        011502  012746  000001                                                          MOV     #1,-(SP)
        011506  010600                                                                  MOV     SP,R0
        011510  104414                                                                  TRAP    C$PNTB
        011512  062706  000004                                                          ADD     #4,SP
 2505   011516  005065  000002          CLR     RETRY(R5)       ;NO RETRIES
 2506   011522  042715  002200          BIC     #BIT10!BIT7,@R5 ;NO RETRY, NORM THRESHOLD
 2507   011526  000460                  BR      CHKRET          ;EXIT
 2508   011530  023765  003326  000002  2$:     CMP     MXRTRY,RETRY(R5)  ;YES. DID WE GRADUATE TO HARD?
 2509   011536  001425                  BEQ     HRD1            ;YES
 2510   011540  005265  000002          INC     RETRY(R5)       ;NO.  JUST ANOTHER
 2511   011544                          PRINTX  #RTRYN,RETRY(R5)  ;PRINT OUT
        011544  016546  000002                                                          MOV     RETRY(R5),-(SP)
        011550  012746  012050                                                          MOV     #RTRYN,-(SP)
        011554  012746  000002                                                          MOV     #2,-(SP)
        011560  010600                                                                  MOV     SP,R0
```

```
                011562  104415                                                                  TRAP    C$PNTX
                011564  062706  000006                                                          ADD     #6,SP
   2512         011570  032715  000200                  BIT     #BIT7,@R5       ;WAS NORMAL THRESHOLD?
   2513         011574  001403                          BEQ     1$              ;YES-REDUCE GAIN
   2514         011576  042715  000200                  BIC     #BIT7,@R5       ;NO-NORM
   2515         011602  000432                          BR      CHKRET
   2516         011604  052715  000200          1$:     BIS     #BIT7,@R5       ;REDUCED
   2517         011610  000427                          BR      CHKRET          ;DONE
   2518         011612  000240                  HRD1:   NOP                     ;HERE IS HARD ERROR!
   2519         011614                                  PRINTX  #UNREC
                011614  012746  012070                                                          MOV     #UNREC,-(SP)
                011620  012746  000001                                                          MOV     #1,-(SP)
                011624  010600                                                                  MOV     SP,R0
                011626  104415                                                                  TRAP    C$PNTX
                011630  062706  000004                                                          ADD     #4,SP
   2520         011634  032715  000400                  BIT     #BIT8,@R5       ;RD OR WR?
   2521         011640  001003                          BNE     4$              ;WRITE
   2522         011642  012704  000016                  MOV     #HRDRD,R4       ;READ
   2523         011646  000402                          BR      5$              ;LOG IT
   2524         011650  012704  000020          4$:     MOV     #HRDWR,R4       ;WRITE
   2525         011654  004737  012500          5$:     CALL    LOG             ;LOG IT
   2526         011660  005065  000002                  CLR     RETRY(R5)       ;BACK TO NORMAL
   2527         011664  042715  002200                  BIC     #BIT10!BIT7,@R5 ;NO RETRY, NOT REDUCED
   2528
   2529         011670  042737  000006  003304  CHKRET: BIC     #BIT1!BIT2,SYSTAT       ;NO SYSTEM RETRY NEXT PASS
   2530         011676  042715  000100                  BIC     #BIT6,@R5       ;NO MORE HOST DATA CHECK ERROR
   2531         011702                                  POP     R4
                011702  012604                                                          MOV     (SP)+,R4

   2532         011704                                  POP     R0
                011704  012600                                                          MOV     (SP)+,R0

   2533         011706  000207                          RETURN
   2534
   2535
   2536         011710      045     101     122  RECOV:  .ASCIZ  /%ARECOVERED FROM DATA CHECK ERROR RETRY # %D1%N/
   2537                                                  .EVEN
   2538         011770      045     101     040  THRSLO: .ASCIZ  /%A NORMAL THRESHOLD%N/
   2539                                                  .EVEN
   2540         012016      045     101     040  THRSHI: .ASCIZ  /%A MODIFIED THRESHOLD %N/
   2541                                                  .EVEN
   2542         012050      045     101     122  RTRYN:  .ASCIZ  /%ARETRY # %D1%N/
   2543                                                  .EVEN
   2544         012070      045     101     125  UNREC:  .ASCIZ  /%AUNRECOVERABLE%N/
   2545                                                  .EVEN
   2546         012112      045     101     117  RETERR: .ASCIZ  /%AOTHER ERROR DURING RETRY : EXIT RETRY%N/
   2547                                                  .EVEN
```

```
                                .SBTTL  CHKSUC / INTERPRET SUCCESS CODE /

                                ;++
                                ; CHKSUC - COPY SUCCESS CODE (BYTE) TO SUCCS+1(R5).  INTERPRET SUCCESS
                                ;          AND IF NOT 0, LOG APPROPRIATE ERROR.
                                ; INPUTS:  R0 POINTS TO END PACKET.
                                ;          @R5 - UNIT STATUS WORD
                                ;          CMDSNT(R5) - COMMAND BYTE
                                ;
                                ; OUTPUTS: R4 IS ERROR NUMBER IF ERROR.
                                ;            SUCCS(R5) UPDATED.
                                ;            BIT9 @R5 SET ON DATA CHECK SUCCESS CODE
                                ;--

012164  000240          CHKSUC:: NOP
012166  016065  000002  000076          MOV     2(R0),SUCCS(R5) ;R0-->END PACKET
012174  122760  000000  000003          CMPB    #ESOK,3(R0)     ;GET SUCCESS BYTE
012202  001535                         BEQ     12$             ;COMPLETE SUCCESS-EXIT

012204  122760  000001  000003          CMPB    #ESTRY,3(R0)    ;OK BUT RETRIES?
012212  001012                         BNE     20$             ;NO
012214  126527  000100  000002          CMPB    CMDSNT(R5),#RSSRD;A READ?
012222  001001                         BNE     22$             ;NO

012224  000520                         BR      10$             ;NO RETRIES IN MAINTENANCE!
012226  126527  000100  000003  22$:    CMPB    CMDSNT(R5),#RSSWR       ;A WRITE?
012234  001001                         BNE     20$             ;NO
012236  000513                         BR      10$             ;LOG IT
012240  122760  177737  000003  20$:    CMPB    #ESNOMO,3(R0)   ;NO MOTOR?
012246  001003                         BNE     1$              ;NO
012250  012704  000030                 MOV     #NOMOT,R4       ;YES-
012254  000506                         BR      11$             ;LOG

012256  122760  177757  000003  1$:     CMPB    #ESCKS,3(R0)    ;"DATA CHECK" ERROR?
012264  001003                         BNE     2$              ;NO
012266  052715  001000                 BIS     #BIT9,@R5       ;SET DATA-CHK-ERROR FLAG
012272  000501                         BR      12$             ;DONT LOG

012274  126527  000100  000007  2$:     CMPB    CMDSNT(R5),#RSSSLF  ;SELF TEST?
012302  001006                         BNE     3$              ;NOPE
012304  105760  000003                 TSTB    3(R0)           ;YES, NEG. IF ERROR
012310  100072                         BPL     12$             ;OK

012312  012704  000044                 MOV     #SLFER,R4       ;YES-ERROR
012316  000465                         BR      11$             ;LOG IT

012320  122760  177740  000003  3$:     CMPB    #ESSK,3(R0)     ;SEEK ERROR?
012326  001005                         BNE     4$              ;NO
012330  012704  000024                 MOV     #SKERR,R4       ;YES-
012334  052705  040000                 BIS     #BIT14,R5       ;SET 'DOBRK' FLAG
012340  000454                         BR      11$             ;LOG

012342  122760  177767  000003  4$:     CMPB    #ESNCRT,3(R0)   ;NO CART?
012350  001003                         BNE     5$              ;NO
012352  012704  000054                 MOV     #NCART,R4       ;YES-
012356  000445                         BR      11$             ;LOG
```

```
 2607 012360  122760  177720  000003  5$:     CMPB    #ESCMD,3(R0)    ;NO UNDERSTAND HOST?
 2608 012366  001003                          BNE     6$              ;NO
 2609 012370  012704  000040                  MOV     #CMNDER,R4      ;YES-
 2610 012374  000436                          BR      11$             ;LOG
 2611
 2612 012376  122760  177770  000003  6$:     CMPB    #ESNONX,3(R0)   ;NON EXISTENT UNIT?
 2613 012404  001003                          BNE     7$              ;NO
 2614 012406  012704  000036                  MOV     #NOUNIT,R4      ;YES-
 2615 012412  000427                          BR      11$             ;LOG
 2616
 2617 012414  122760  177765  000003  7$:     CMPB    #ESWLOC,3(R0)   ;WRITE LOCKED?
 2618 012422  001003                          BNE     8$              ;NO
 2619 012424  012704  000026                  MOV     #WRLOCK,R4      ;YES-
 2620 012430  000420                          BR      11$             ;LOG
 2621
 2622 012432  122760  177776  000003  8$:     CMPB    #ESPART,3(R0)   ;PARTIAL OP?
 2623 012440  001003                          BNE     9$              ;NO
 2624 012442  012704  000034                  MOV     #PARTL,R4       ;YES-
 2625 012446  000411                          BR      11$             ;LOG
 2626
 2627 012450  122760  177711  000003  9$:     CMPB    #ESREC,3(R0)    ;WRONG RECORD?
 2628 012456  001003                          BNE     10$             ;NO
 2629 012460  012704  000042                  MOV     #RECERR,R4      ;YES-
 2630 012464  000402                          BR      11$             ;LOG
 2631
 2632 012466  012704  000046          10$:    MOV     #SUCOTL,R4      ;UNDEFINED
 2633 012472  004737  012500          11$:    CALL    LOG             ;LOG ERROR
 2634 012476  000207                  12$:    RETURN                  ;RETURN
```

```
                                .SBTTL  LOG / TO LOG ERROR IN CORRECT PLACE

                                ;++
                                ; LOG - DETERMINE IF ERROR IS FATAL, NON-FATAL OR FATAL AFTER N TRIES
                                ;       BY INDEX (ERROR #) INTO DEVICE DATA BLOCK.  ADD THE DRIVE # TO
                                ;       INDICATE UPPER OR LOWER BYTE AND INCREMENT THAT ERROR UNLESS
                                ;       THAT BYTE WOULD OVERFLOW.  DETERMINE IF EVL FLAG SET,AND IF SO,
                                ;       CHECK THRESHOLD (EVLTHR) AND  PRINT APPROPRIATE ERROR MESSAGE
                                ;       DESCRIPTION.  ABORT THE UNIT IF INDICATED THROUGH DODROP CODE.
                                ; INPUTS:  R4 = ERROR CODE
                                ; OUTPUTS: ABNDX(R5) = ERROR CODE.
                                ;           DLV(R5) = 0
                                ;          L$LUN = UNIT NUMBER
                                ;--

012500                          LOG::   PUSH    R0
012500  010046                                          MOV     R0,-(SP)

012502                                  PUSH    R1
012502  010146                                          MOV     R1,-(SP)

012504                                  PUSH    R3
012504  010346                                          MOV     R3,-(SP)

012506                                  PUSH    R4
012506  010446                                          MOV     R4,-(SP)


012510  011537  002074                  MOV     @R5,L$LUN       ;GET UNIT NUMBER
012514  042737  177770  002074          BIC     #177770,L$LUN   ;MASK IT OFF
012522  010465  000004                  MOV     R4,ABNDX(R5)    ;SAVE INDEX IN CASE OF ABORT MESSAGE
012526  012703  000120                  MOV     #LGOFST,R3      ;OFFSET TO LOW ORDER BYTE (DRIVE0)
012532  060403                          ADD     R4,R3           ;FORM INDEX OF PARAM. TO UPDATE
012534  060503                          ADD     R5,R3           ;FORM ABSOLUTE ADDR. THIS UNIT
012536  004737  013504                  CALL    WHCHDR          ;SEE WHICH DRIVE T'WAS
012542  103001                          BCC     2$              ;WAS DRIVE 0
012544  005203                          INC     R3              ;DRIVE 1; POINT TO UPPER BYTE
012546  122713  000377          2$:     CMPB    #255.,@R3       ;POTENTIAL OVERFLOW POSSIBLE?
012552  001005                          BNE     LOGOK           ;NO
012554                          LOG0:   ERRDF   0.,OVRFLO,ERRDES  ;YES
012554  104455                                                                  TRAP    C$ERDF
012556  000000                                                                  .WORD   0
012560  013400                                                                  .WORD   OVRFLO
012562  013034                                                                  .WORD   ERRDES
012564  000512                          BR      ABO             ;ABORT UNIT
012566  105213                  LOGOK:  INCB    @R3             ;INCREMENT THE ERROR
012570  111304                          MOVB    @R3,R4          ;TEMP'LY SAVE IT
012572  016503  000004                  MOV     ABNDX(R5),R3    ;GET INDEX AGAIN
012576  012701  002224                  MOV     #RSNTAB,R1      ;FORM ADRS OF MSG
012602  066501  000004                  ADD     ABNDX(R5),R1    ;LIKE THIS
012606  042701  000001                  BIC     #BIT0,R1        ;INSURE WORD BOUNDARY
012612  032737  000004  016616          BIT     #EVL,FLGLOC     ;EVL SELECTED?
012620  001414                          BEQ     LOGOK2          ;NO-CONT
```

```
                                                CMPB    EVLTHR,R4       ;YES,OVER THRESHOLD?
                                                BHI     LOGOK2          ;NO
                                                MOV     R3,DFTL1+2      ;YES,LOAD ERROR #
                                                MOV     @R1,DFTL1+4     ;AND MESSAGE ADDR
                                        DFTL1:  ERRDF   0,DFTL1,ERRDES  ;ERROR
012640 104455                                                                   TRAP    C$ERDF
012642 000000                                                                   .WORD   0
012644 012640                                                                   .WORD   DFTL1
012646 013034                                                                   .WORD   ERRDES
                                                BR      ABO             ;DROP IT
012652 120327 000014                    LOGOK2: CMPB    R3,#BDCOM       ;'NEVER FATAL' TYPE?
012656 103011                                   BHIS    NTSFT           ;NO
012660 010337 012672                            MOV     R3,LOG1+2       ;YES, ERROR CODE
012664 011137 012674                            MOV     @R1,LOG1+4      ;DESCRIPTION
012670                                  LOG1:   ERRSOFT 0.,LOG1,ERRDES
012670 104457                                                                   TRAP    C$ERSOFT
012672 000000                                                                   .WORD   0
012674 012670                                                                   .WORD   LOG1
012676 013034                                                                   .WORD   ERRDES
012700 000450                                   BR      LOGO            ;EXIT

012702 120327 000026                    NTSFT:  CMPB    R3,#WRLOCK      ;ONE TRY?
012706 103411                                   BLO     MABEE           ;NO, MAYBE A MULTIPLE
012710 010337 012722                            MOV     R3,LOG2+2.      ;YES
012714 011137 012724                            MOV     @R1,LOG2+4
012720                                  LOG2:   ERRHRD  0,LOG2,ERRDES   ;PRINT HARD MESSAGE
012720 104456                                                                   TRAP    C$ERHRD
012722 000000                                                                   .WORD   0
012724 012720                                                                   .WORD   LOG2
012726 013034                                                                   .WORD   ERRDES
012730 000430                                   BR      ABO             ;DROP UNIT

012732 042704 177400                    MABEE:  BIC     #177400,R4      ;NEGATE SIGN EXTEND
012736 163704 003316                    1$:     SUB     FTLNM,R4        ;SEE IF MULTIPLE OF
012742 001413                                   BEQ     HRD             ;FTLNM-YES!
012744 103401                                   BLO     SFT             ;NO
012746 000773                                   BR      1$              ;NOT THERE YET

012750 010337 012762                    SFT:    MOV     R3,LOG3+2       ;ERROR CODE
012754 011137 012764                            MOV     @R1,LOG3+4      ;DESCRIPTION
012760                                  LOG3:   ERRSOFT 0,LOG3,ERRDES
012760 104457                                                                   TRAP    C$ERSOFT
012762 000000                                                                   .WORD   0
012764 012760                                                                   .WORD   LOG3
012766 013034                                                                   .WORD   ERRDES
012770 000414                                   BR      LOGO            ;EXIT
012772 010337 013004                    HRD:    MOV     R3,LOG3B+2      ;HARD ERROR CODE
012776 011137 013006                            MOV     @R1,LOG3B+4     ;DESCRIPTION
013002                                  LOG3B:  ERRHRD  0,LOG3B,ERRDES
013002 104456                                                                   TRAP    C$ERHRD
013004 000000                                                                   .WORD   0
013006 013002                                                                   .WORD   LOG3B
013010 013034                                                                   .WORD   ERRDES

013012 011500                           ABO:    MOV     @R5,R0          ;GET UNIT NUMBER
013014 042700 177770                            BIC     #177770,R0      ;UN-SIGN EXTEND
013020                                          DODU    R0              ;USE LOGICAL # TO DROP
```

```
                013020  104451                                                  TRAP    C$DODU
2715            013022                  LOGO:   POP     R4              ;RESTORE
                013022  012604                                          MOV     (SP)+,R4

2716            013024                          POP     R3
                013024  012603                                          MOV     (SP)+,R3

2717            013026                          POP     R1
                013026  012601                                          MOV     (SP)+,R1

2718            013030                          POP     R0
                013030  012600                                          MOV     (SP)+,R0

2719            013032  000207                  RETURN                  ;RETURN
```

```
                                ;++
                                ; ERRDES - CONTAINS CODE FOR EXTENDED ERROR INFORMATION:  DRIVE #,
                                ;          BLOCK #, ETC.
                                ;--

                                        BGNMSG  ERRDES          ;ERROR DESCRIPTION
                                                                                ERRDES::
                                        PUSH    R0
                                                                MOV     R0,-(SP)

                                        PUSH    R2
                                                                MOV     R2,-(SP)

                                        CLR     R2              ;PRESET TO DATA TYPE
                                        BIT     #BIT4,@R5       ;WHAT PACK TYPE?
                                        BEQ     2$              ;DATA
                                        INC     R2              ;COMMAND
                                2$:     PRINTB  #UNIT,<B,DR(R5)>,R2,<B,SYSTAT+1>
                                                                                CLR     -(SP)
                                                                                BISB    SYSTAT+1,(SP)
                                                                                MOV     R2,-(SP)
                                                                                CLR     -(SP)
                                                                                BISB    DR(R5),(SP)
                                                                                MOV     #UNIT,-(SP)
                                                                                MOV     #4,-(SP)
                                                                                MOV     SP,R0
                                                                                TRAP    C$PNTB
                                                                                ADD     #12,SP
                                        MOV     REC(R5),R0      ;RECORD NUMBER
                                        MOV     PATTEN(R5),R2   ;DATA EXPECTED
                                        PRINTB  #RECID,R0,<B,CMDSNT(R5)>,<B,R2>,<B,SUCCS+1(R5)>
                                                                                CLR     -(SP)
                                                                                BISB    SUCCS+1(R5),(SP)
                                                                                CLR     -(SP)
                                                                                BISB    R2,(SP)
                                                                                CLR     -(SP)
                                                                                BISB    CMDSNT(R5),(SP)
                                                                                MOV     R0,-(SP)
                                                                                MOV     #RECID,-(SP)
                                                                                MOV     #5,-(SP)
                                                                                MOV     SP,R0
                                                                                TRAP    C$PNTB
                                                                                ADD     #14,SP
                                        TST     DLV(R5)         ;DLV ERROR?
                                        BEQ     3$              ;NO
                                        PRINTB  #RECID2,DLV(R5) ;YES-PRINT
                                                                                MOV     DLV(R5),-(SP)
                                                                                MOV     #RECID2,-(SP)
                                                                                MOV     #2,-(SP)
                                                                                MOV     SP,R0
                                                                                TRAP    C$PNTB
                                                                                ADD     #6,SP
                                        CLR     DLV(R5)         ;RESET
                                3$:     POP     R2              ;RESTORE
                                                                MOV     (SP)+,R2
```

```
2743 013222                                    POP     R0
     013222  012600                                    MOV     (SP)+,R0

2744 013224                                    ENDMSG                  ;EXIT
     013224                                                                    L10003:
     013224  104423                                                            TRAP    C$MSG
2745 013226     045     101     104  UNIT::   .ASCIZ  /%ADRIVE#  %01%A  PAK SENT %01%A FLAG RCVD %03%N/
2746                                           .EVEN
2747 013306     045     101     102  RECID::  .ASCIZ  /%ABLOCK#  %04%A COMMAND %02%A EXPCTD %03%A SUCCESS %03%N/
2748                                           .EVEN
2749 013400     103     101     116  OVRFLO:  .ASCIZ  /CAN'T UPDATE ERROR OR STATISTIC:OVERFLOW PENDING/
2750                                           .EVEN
2751 013462     045     101     040  RECID2:  .ASCIZ  /%A RCDB WAS %06%N/
2752                                           .EVEN
```

```
                                .SBTTL  WHCHDR / SEE WHICH DRIVE IS ACTIVE

                                ;++
                                ; INPUTS:  DR(R5)
                                ; OUTPUTS:  CARRY=DRIVE (1 OR 0)
                                ;--

013504  000241                  WHCHDR:: CLC                    ;CLEAR CARRY

013506  105765  000060                 TSTB    DR(R5)         ;DR 0?
013512  001401                       BEQ     2$             ;YES
013514  000261                       SEC                    ;NO

013516  000207                  2$:     RETURN                 ;RETURN
```

```
                                .SBTTL  CHKSUM / FORM THE PACKET CHECKSUM

                                ;++
                                ; THE CHECKSUM IS A 16 BIT CHECKSUM WITH END-AROUND CARRY.
                                ;
                                ; INPUTS:  R0 -> (POINTS TO) TOP OF PACKET
                                ;          R1 = # OF BYTES
                                ; OUTPUTS:  R0 -> WHERE TO PUT CHECKSUM
                                ;           R1 = CHECKSUM
                                ;--

013520                          CHKSUM:: PUSH   R3
013520  010346                                   MOV     R3,-(SP)

013522                                  PUSH    R2
013522  010246                                   MOV     R2,-(SP)

013524  042737  000001  003304          BIC     #BIT0,SYSTAT    ;"CHECKSUM IS ODD" BIT
013532  032701  000001                  BIT     #BIT0,R1        ;AN ODD # OF BYTES?
013536  001403                          BEQ     1$              ;NO
013540  052737  000001  003304          BIS     #BIT0,SYSTAT    ;YES

013546  006001                  1$:     ROR     R1              ;/2 FOR WORDS

013550  005003                  2$:     CLR     R3              ;PREP CHECKSUM WORD

013552  062003                  3$:     ADD     (R0)+,R3        ;FORM SUM
013554  005503                          ADC     R3              ;WITH CARRY
013556  005301                          DEC     R1              ;MORE WORDS?
013560  001374                          BNE     3$              ;YES

013562  032737  000001  003304          BIT     #BIT0,SYSTAT    ;WAS IT ODD
013570  001405                          BEQ     4$              ;NO
013572  112002                          MOVB    (R0)+,R2        ;YES GET NEXT BYTE
013574  042702  177400                  BIC     #177400,R2      ;UN-SIGN EXTEND
013600  060203                          ADD     R2,R3           ;ADD IT IN
013602  005503                          ADC     R3              ;AND CARRY JUST IN CASE

013604  010301                  4$:     MOV     R3,R1           ;RETURN IT IN CORRECT PLACE
013606                                  POP     R2              ;RESTORE
013606  012602                                   MOV     (SP)+,R2

013610                                  POP     R3
013610  012603                                   MOV     (SP)+,R3

013612  000207                          RETURN                  ;RETURN
```

```
                                .SBTTL  CKCKSM / MODULE TO CHECK THE CHKSUMS

                                ;++
                                ; MAKE SURE THE CHECKSUM RECEIVED = THE CHECKSUM CALCULATED.
                                ; INPUTS:  R4 = THE PACKET BYTE COUNT
                                ;          R0 -> THE PACKET TOP
                                ; OUTPUTS:  CARRY SET IF CHECKSUM CALC'D DOES NOT EQUAL CHECKSUM SENT
                                ;           R0 -> THE PACKET TOP
                                ;--


013614                          CKCKSM:: PUSH   R1
013614  010146                                                 MOV     R1,-(SP)

013616                                  PUSH    R0              ;SAVE
013616  010046                                                 MOV     R0,-(SP)

013620  010401                          MOV     R4,R1           ;COPY BYTE COUNT TO CORRECT
013622  004737  013520                 CALL    CHKSUM          ;REGISTER FOR CHKSUM AND
                                                                ;FORM CHECKSUM

                                ;HERE R0 --> XMITTED CHKSUM, R1=CHKSUM CALC'D

013626  122001                          CMPB    (R0)+,R1        ;LOWER ORDER CHECK
013630  001005                          BNE     2$              ;WRONG

013632  000301                          SWAB    R1              ;OK-PREP FOR

013634  122001                          CMPB    (R0)+,R1        ;HIGH ORDER CHECK
013636  001002                          BNE     2$              ;WRONG
013640  000241                          CLC                     ;OK-CLEAR SAILING

013642  000401                          BR      3$              ;EXIT

013644  000261                  2$:     SEC                     ;LET ERROR BE KNOWN


013646                          3$:     POP     R0
013646  012600                                                 MOV     (SP)+,R0

013650                                  POP     R1
013650  012601                                                 MOV     (SP)+,R1

013652  000207                          RETURN                  ;RETURN
```

```
                                .SBTTL  DOBRK / MODULE TO INIT TU58 AND TEST INTERRUPTS

                                ;++
                                ; DOBRK - SEND RADIAL SERIAL "BREAK" TO DEVICE:
                                ;       - SET "BREAK" ON INTERFACE.
                                ;       - SEND 8. NULLS
                                ;       - CLEAR "BREAK" ON INTERFACE
                                ;       - SET VECTORS FOR RCV AND XMIT
                                ;       - SEND 2 BYTES OF "INIT"
                                ;       - RECEIVE "CONTINUE"
                                ;       - IF RECEIVE GARBAGE OR TIMEOUT - ERROR
                                ;       - CLEAR INTERRUPTS AND VECTORS
                                ; INPUTS:  @R5 BIT14 WAS SET - (SEND BREAK)
                                ; OUTPUTS: @R5 BIT14 CLEAR IF SUCCESSFUL INIT.
                                ;          SYSTAT+1 = RECEIVED BYTE
                                ;          ERRORS R4 = ERROR CODE:
                                ;               - SEND NOT READY TIMEOUT (TOSNDB)
                                ;               - NO RESPONSE
                                ;               - DLV ERROR
                                ;               - CAN'T INIT
                                ;--

013654  105037  014407          DOBRK:: CLRB    INITWD+1        ;CLEAR BYTE RECEIVE ADDR
013660  005037  014410                  CLR     BRKTO           ;CLEAR TIME OUT CONSTANT
013664  052775  000001  000026        BIS     #BIT0,@XMSR(R5) ;SET 'BREAK'
013672  012765  000001  000100        MOV     #RSSNIT,CMDSNT(R5)  ;SAY WE SENT 'INIT'
013700  052715  000020                BIS     #BIT4,@R5       ;PAK SENT TYPE =COMMAND, SORT OF
013704  012704  000010                MOV     #8.,R4          ;BREAK-IT'S-BACK COUNT=8
013710                          1$:     BREAK                   ;SUPERVISOR TAKE FIVE
013710  104422                                                                  TRAP    C$BRK
                                                                ;FOR ^C CHECK, ETC.
013712  105775  000026                TSTB    @XMSR(R5)       ;READY?
013716  100410                      BMI     4$              ;YES
013720  005337  014410              DEC     BRKTO           ;NO,TIME OUT?
013724  001371                      BNE     1$              ;NO
013726  012704  000056              MOV     #TOSNDB,R4      ;YES, SET ERROR CODE
013732  004737  012500              CALL    LOG             ;LOG IT
013736  000535                      BR      3$              ;EXIT
013740  113775  014404  000030  4$: MOVB    BRKWD,@XMDB(R5) ;SEND NULL
013746  005037  014410              CLR     BRKTO           ;RESET TIME OUT
013752  005304                      DEC     R4              ;MORE NULLS TO SEND?
013754  001355                      BNE     1$              ;YES
013756  005375  000026              DEC     @XMSR(R5)       ;NO, CLEAR 'BREAK'
013762  017500  000024              MOV     @RCDB(R5),R0    ;HEAVE 'GARBAGE' 1ST BYTE
013766                          SETPRI  #PRI00          ;SET TO INTERRUPT FO SURE
013766  012700  000000                                                  MOV     #PRI00,R0
013772  104441                                                          TRAP    C$SPRI
013774                          SETVEC  TUVECT(R5),#RCVINT,#PRI07 ;SET VECTO INFO
013774  012746  000340                                                  MOV     #PRI07,-(SP)
014000  012746  014314                                                  MOV     #RCVINT,-(SP)
014004  016546  000204                                                  MOV     TUVECT(R5),-(SP)
014010  012746  000003                                                  MOV     #3,-(SP)
014014  104437                                                          TRAP    C$SVEC
014016  062706  000010                                                  ADD     #10,SP
014022  062765  000004  000204        ADD     #4,TUVECT(R5)   ;AND INC TO SND VECTOR
014030                          SETVEC  TUVECT(R5),#SNDINT,#PRI07;AND SET IT
014030  012746  000340                                                  MOV     #PRI07,-(SP)
```

```
        014034  012746  014300                                                          MOV     #SNDINT,-(SP)
        014040  016546  000204                                                          MOV     TUVECT(R5),-(SP)
        014044  012746  000003                                                          MOV     #3,-(SP)
        014050  104437                                                                  TRAP    C$SVEC
        014052  062706  000010                                                          ADD     #10,SP
  2898  014056  162765  000004  000204        SUB     #4,TUVECT(R5)   ;RESET VECTOR ADDR.
  2899  014064  005037  014410                CLR     BRKTO           ;RESET TIME OUT
  2900  014070  012704  014406                MOV     #INITWD,R4      ;USE ADDR. FOR SNDBYT
  2901  014074  010437  014412                MOV     R4,BRKPTR       ;AND SAVE FOR 'WAIT'
  2902  014100  052775  000100  000026        BIS     #BIT6,@XMSR(R5) ;ENABLE INTER.
  2903  014106  004737  014350                CALL    WAIT            ;AND ENTER LOOP
  2904  014112  005715                        TST     @R5             ;ABORTED FROM TIME OUT?
  2905  014114  100446                        BMI     3$              ;YES-EXIT
  2906
  2907  014116  005037  014410                CLR     BRKTO           ;RESET TIME OUT
  2908  014122  012704  014406                MOV     #INITWD,R4      ;SEND SECOND INIT
  2909  014126  010437  014412                MOV     R4,BRKPTR       ;SAVE POINTER AGAIN
  2910  014132  052775  000100  000026        BIS     #BIT6,@XMSR(R5) ;AND THEN ENABLE INT
  2911  014140  004737  014350                CALL    WAIT            ;AND WAIT
  2912  014144  005715                        TST     @R5             ;IF ABORTED
  2913  014146  100431                        BMI     3$              ;THEN EXIT
  2914
  2915  014150  012704  014407                MOV     #INITWD+1,R4    ;WHERE RESPONSE WILL GO (ADDRESS)
  2916  014154  010437  014412                MOV     R4,BRKPTR       ;AND FOR 'WAIT'
  2917  014160  052775  000100  000022        BIS     #BIT6,@RCSR(R5) ;ENABLE RECIEVE INT.
  2918  014166  004737  014350                CALL    WAIT            ;GET ANSWER
  2919  014172  005715                        TST     @R5             ;ABORTED?
  2920  014174  100416                        BMI     3$              ;YES.
  2921
  2922  014176  123727  014407  000020        CMPB    INITWD+1,#RSCONT  ;NO, IS IT 'CONTINUE'?
  2923  014204  001003                        BNE     2$              ;NOPE-ERROR
  2924
  2925  014206  042715  040000                BIC     #BIT14,@R5      ;SUCCESSFUL, CLEAR DOBREAK FLAG
  2926  014212  000407                        BR      3$              ;EXIT
  2927
  2928  014214  113737  014407  003305  2$:   MOVB    INITWD+1,SYSTAT+1       ;SAVE BUM RESPONSE
  2929  014222  012704  000032                MOV     #CNINIT,R4              ;CAN'T INIT CODE
  2930  014226  004737  012500                CALL    LOG             ;LOG IT
  2931                                                                ;SCHEDULER WILL TRY AGAIN IF NOT ABORTED
  2932
  2933  014232  042775  000100  000026  3$:   BIC     #BIT6,@XMSR(R5) ;CLEAR INTERRUPTS
  2934  014240  042775  000100  000022        BIC     #BIT6,@RCSR(R5) ; AND FOR RECIEVE
  2935  014246                                CLRVEC  TUVECT(R5)      ;RELEASE RECIEVE VECT.
        014246  016500  000204                                                          MOV     TUVECT(R5),R0
        014252  104436                                                                  TRAP    C$CVEC
  2936  014254  062765  000004  000204        ADD     #4,TUVECT(R5)   ;AND GET SEND ADDR.
  2937  014262                                CLRVEC  TUVECT(R5)      ;AND RELEASE IT
        014262  016500  000204                                                          MOV     TUVECT(R5),R0
        014266  104436                                                                  TRAP    C$CVEC
  2938  014270  162765  000004  000204        SUB     #4,TUVECT(R5)   ;RESTORE POINTER
  2939  014276  000207                        RETURN                  ;RETURN
```

```
                                .SBTTL  INTERRUPT SERVICE ROUTINES AND TIMER
                        BGNSRV  SNDINT                  ;"SEND" INTERRUPT SERVICE:
014300                                                                  SNDINT::
                        SNDHND: BIC     #BIT6,@XMSR(R5) ;DISABLE INTERRUPT
                                MOVB    (R4)+,@XMDB(R5);OUTPUT BYTE
                        ENDSRV
014312                                                                  L10004:
014312  000002                                                                RTI
                        BGNSRV  RCVINT                  ;"RCV" INTERRUPT SERVICE:
014314                                                                  RCVINT::
                        RCVHND: BIC     #BIT6,@RCSR(R5) ;DISABLE INTS
                                MOV     @RCDB(R5),DLV(R5)       ;SAVE WORD
                                MOVB    DLV(R5),(R4)+   ;BYTE TO BUFFER
                                TST     DLV(R5)         ;ERROR?
                                BMI     10$             ;YES
                                CLR     DLV(R5)         ;NO CLEAR ERROR
                        10$:
                        ENDSRV
014346                                                                  L10005:
014346  000002                                                                RTI


                WAIT:   NOP                             ;WAIT  LOOP FOR
                                                        ;INTERRUPT SERVICING
                        CMP     R4,BRKPTR               ;IF=,THEN NO INTERRUPT
                        BNE     1$                      ;GOT ONE!
                        BREAK                           ;SUPERVISOR BREAK
014360  104422                                                          TRAP    C$BRK
                        BREAK                           ;KILL SOME TIME
014362  104422                                                          TRAP    C$BRK
                        DEC     BRKTO                   ;TIME OUT?
                        BNE     WAIT                    ;NO...CONT.
                        MOV     #TORCVB,R4              ;YES LOAD ERROR #
                        CALL    LOG                     ;LOG IT
                1$:     RETURN                          ;RETURN

                BRKWD:  .WORD   0                       ;NULL
                INITWD: .BYTE   RSINIT                  ;INIT COMMAND
                        .BYTE   0                       ;RSCONT IS EXPECTED HERE
                BRKTO:  .WORD   0                       ;TIME OUT
                BRKPTR: .WORD   0                       ;POINTER TO INITWD
```

```
                                .SBTTL  COMPAR/DATA COMPARISON MODULE

                                ;++
                                ; COMPAR - IF "COMPARE DATA" SELECTED, COMPARE EACH DATA BYTE OF PACKET
                                ;          TO PATTEN(R5).  SAVE NUMBER OF BYTES NOT CORRECT.  IF NOT
                                ;          0, PRINT SOFT ERROR AND TOTAL # WRONG BYTES.  SET 'BAD_DATA_
                                ;          IN_PACKET" BIT (BIT6 @R5) FOR HIGHER LEVEL MODULES.
                                ; INPUTS:  - ?CMPDAT) FLAG TO NOT COMPARE (=1)
                                ;          - PKPTR(R5) POINTS TO DATA PACK.
                                ; OUTPUTS: BIT6 @R5 (BAD DATA FLAG) ADJUSTED.
                                ;          L$LUN - UNIT NUMBER
                                ;          PRNSIZ - SIZE OF PACKET
                                ;--

014414                          COMPAR:: PUSH   R0              ;COMPARE DATA IS DATA PACKET
014414  010046                                                  MOV     R0,-(SP)

014416                                   PUSH   R4              ;TO PATTERN WRITTEN
014416  010446                                                  MOV     R4,-(SP)

014420                                   PUSH   R1              ;USING BYTE COUNT IN PACKET
014420  010146                                                  MOV     R1,-(SP)

014422  005037  014572                     CLR    BDBYTS          ;CLEAR TOTAL WRONG
014426  016504  000104                     MOV    PKPTR(R5),R4    ;GET TOP OF PACKET
014432  005737  002214                     TST    CMPDAT          ;COMPARE SELECTED?
014436  001451                           BEQ    4$              ;NO-EXIT
014440  005204                           INC    R4              ;YES, LOCATE COUNT
014442  111401                           MOVB   @R4,R1          ;GET IT
014444  042701  177400                  BIC    #177400,R1      ;SIGN-UNEXTEND
                                                                ;MUST TEST BYTE-WISE...
014450  005204                           INC    R4              ;-->FIRST DATA  BYTE
014452  126524  000072          1$:      CMPB   PATTEN(R5),(R4)+  ;DATA-WHAT WAS EXPECTED?
014456  001402                           BEQ    2$              ;YES
014460  005237  014572                     INC    BDBYTS          ;NO, INCREMENT TOTAL WRONG
014464  005301                    2$:      DEC    R1              ;MORE LEFT?
014466  001371                           BNE    1$              ;YES
014470  005737  014572                     TST    BDBYTS          ;ANY WRONG?
014474  001432                           BEQ    4$              ;NO
014476  011537  002074                     MOV    @R5,L$LUN       ;GET UNIT NUMBER
014502  042737  177770  002074           BIC    #177770,L$LUN   ;MASK IT OFF
014510                                   ERRSOFT 0.,MSBDA,ERRDES  ;YES-PRINT 'BAD DATA IN PACKET' ERROR
014510  104457                                                          TRAP    C$ERSOFT
014512  000000                                                          .WORD   0
014514  002336                                                          .WORD   MSBDA
014516  013034                                                          .WORD   ERRDES
014520                                   PRINTB #DESC,BDBYTS
014520  013746  014572                                                  MOV     BDBYTS,-(SP)
014524  012746  014574                                                  MOV     #DESC,-(SP)
014530  012746  000002                                                  MOV     #2,-(SP)
014534  010600                                                          MOV     SP,R0
014536  104414                                                          TRAP    C$PNTB
014540  062706  000006                                                  ADD     #6,SP
014544  052715  000100                   BIS    #BIT6,@R5       ;LET 'EM KNOW UPSTAIRS-BAD DATA FLAG
```

```
3022 014550  012737  000204  003334          MOV     #132.,PRNSIZ    ;SIZE IS ONE DATA PACK
3023 014556  004737  014630                  CALL    PRNPAK          ;AND PRINT THE PACKET
3024 014562                          4$:     POP     R1              ;RESTORE
     014562  012601                                                  MOV     (SP)+,R1

3025 014564                                  POP     R4
     014564  012604                                                  MOV     (SP)+,R4

3026 014566                                  POP     R0
     014566  012600                                                  MOV     (SP)+,R0

3027
3028 014570  000207                          RETURN
3029
3030 014572  000000                  BDBYTS: .WORD
3031 014574     045      101      124 DESC:   .ASCIZ  /%ATOTAL BAD BYTES= %D3%A.%N/
3032                                         .EVEN
```

```
                                .SBTTL  PRNPAK/MODULE TO PRINT DATA PACKET

                                ;++
                                ; PRNPAK - IF PRINT_DATA_PACK_ON_ERROR SELECTED: PRINT EACH BYTE OF PACKET
                                ;          TO BY PKPTR(R5).
                                ; INPUTS:  PRNSIZ - # OF BYTES IN PACKET.
                                ; OUTPUTS: NONE
                                ;--

014630  000240                  PRNPAK:: NOP                    ;PRINTS 1 PACKET
                                                                ;PKPTR(R5)->TOP OF PACKET
                                                                ;PRNSIZ (PASSED)=BYTE COUNT
014632                                  PUSH    R0
014632  010046                                                  MOV     R0,-(SP)

014634                                  PUSH    R4
014634  010446                                                  MOV     R4,-(SP)

014636  105737  002212                  TSTB    PRBUF           ;PRINT PACKET SELECTED?
014642  001451                          BEQ     4$              ;NO
014644  016504  000104                  MOV     PKPTR(R5),R4    ;YES-GET TOP OF PACK
014650  012737  000020  014774  1$:     MOV     #16.,LNCNT      ;16 BYTES PER LINE
014656  112437  014776          2$:     MOVB    (R4)+,PRDAT     ;AVOID SIGN EXTEND
014662                                  PRINTF  #PRFORM,<B,PRDAT>  ;PRINT BYTE
014662  005046                                                          CLR     -(SP)
014664  153716  014776                                                  BISB    PRDAT,(SP)
014670  012746  015000                                                  MOV     #PRFORM,-(SP)
014674  012746  000002                                                  MOV     #2,-(SP)
014700  010600                                                          MOV     SP,R0
014702  104417                                                          TRAP    C$PNTF
014704  062706  000006                                                  ADD     #6,SP
014710  005337  003334                  DEC     PRNSIZ          ;ONE LESS
014714  001414                          BEQ     3$              ;NO MORE
014716  005337  014774                  DEC     LNCNT           ;NEW LINE?
014722  001355                          BNE     2$              ;NOT YET
014724                                  PRINTF  #CARLF          ;YES
014724  012746  015010                                                  MOV     #CARLF,-(SP)
014730  012746  000001                                                  MOV     #1,-(SP)
014734  010600                                                          MOV     SP,R0
014736  104417                                                          TRAP    C$PNTF
014740  062706  000004                                                  ADD     #4,SP
014744  000741                          BR      1$              ;NEXT LINE
014746                          3$:     PRINTF  #CARLF          ;FINISH UP
014746  012746  015010                                                  MOV     #CARLF,-(SP)
014752  012746  000001                                                  MOV     #1,-(SP)
014756  010600                                                          MOV     SP,R0
014760  104417                                                          TRAP    C$PNTF
014762  062706  000004                                                  ADD     #4,SP
014766                          4$:     POP     R4
014766  012604                                                  MOV     (SP)+,R4

014770                                  POP     R0
014770  012600                                                  MOV     (SP)+,R0

014772  000207                  RETURN                          ;RETURN
```

```
GLOBAL AREAS    MACRO M1113  25-SEP-81 10:06  PAGE 88-1                    SEQ 0081
PRNPAK/MODULE TO PRINT DATA PACKET

 3065
 3066 014774  000000                    LNCNT:  .WORD
 3067 014776  000000                    PRDAT:  .WORD
 3068 015000     045     117     063    PRFORM: .ASCIZ  /%03%A /
 3069                                           .EVEN
 3070 015010     045     116     000    CARLF:  .ASCIZ  /%N/
 3071                                           .EVEN
 3072
 3073 015014                                    ENDMOD
 3074
```

```
                                .TITLE MISCELLANEOUS SECTIONS
                                .SBTTL  REPORT CODING SECTION

                                        BGNMOD

                                ;++
                                ; THE RFPORT CODING SECTION CONTAINS THE
                                ; 'PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
                                ;--

                                        BGNRPT
015014                                                                          L$RPT::
015014                                  PUSH    R0
015014  010046                                          MOV     R0,-(SP)

015016                                  PUSH    R1
015016  010146                                          MOV     R1,-(SP)

015020                                  PUSH    R2
015020  010246                                          MOV     R2,-(SP)

015022                                  PUSH    R3
015022  010346                                          MOV     R3,-(SP)

015024                                  PUSH    R4
015024  010446                                          MOV     R4,-(SP)

015026                                  PUSH    R5
015026  010546                                          MOV     R5,-(SP)


015030                                  BREAK
015030  104422                                                          TRAP    C$BRK
015032  012737  003346  015442          MOV     #BLKTBL,RPTR    ;GET 1ST DEVICE BLOCK
015040                                  PRINTS  #STATHD         ;HEADER
015040  012746  015444                                                  MOV     #STATHD,-(SP)
015044  012746  000001                                                  MOV     #1,-(SP)
015050  010600                                                          MOV     SP,R0
015052  104416                                                          TRAP    C$PNTS
015054  062706  000004                                                  ADD     #4,SP
015060                                  BREAK                   ;^C CHECK
015060  104422                                                          TRAP    C$BRK
015062                                  PRINTS  #STHD2          ;2ND HEADER
015062  012746  015720                                                  MOV     #STHD2,-(SP)
015066  012746  000001                                                  MOV     #1,-(SP)
015072  010600                                                          MOV     SP,R0
015074  104416                                                          TRAP    C$PNTS
015076  062706  000004                                                  ADD     #4,SP
015102                          1$:     BREAK                   ;^C CHECK
015102  104422                                                          TRAP    C$BRK
015104  017705  000332                  MOV     @RPTR,R5        ;GET DEVICE BLOCK
```

```
   3139 015110  032715  004000                BIT     #BIT11,@R5      ;UNIT NOT TESTED?
   3140 015114  001131                        BNE     2$              ;TRUE, DON'T PRINT STATISTICS
   3141                                                               ;OK TO PRINT
   3142 015116  011537  015440                MOV     @R5,RLUN        ;SAVE STATUS WORD
   3143 015122  042737  177770  015440        BIC     #177770,RLUN    ;MASK UNIT NUM.
   3144 015130  116501  000122                MOVB    SOFTR(R5),R1    ;SOFTREAD
   3145 015134  042701  177400                BIC     #177400,R1      ;SIGN-UNEXTEND
   3146 015140  116502  000124                MOVB    SOFTW(R5),R2    ;SOFT WRITE
   3147 015144  042702  177400                BIC     #177400,R2      ;
   3148 015150  116503  000136                MOVB    HARDR(R5),R3    ;HARD READ
   3149 015154  042703  177400                BIC     #177400,R3      ;
   3150 015160  116504  000140                MOVB    HARDW(R5),R4    ;HARD WRITE
   3151 015164  042704  177400                BIC     #177400,R4      ;
   3152 015170                                PRINTS  #FM0,RLUN       ;SUMMARY/UNIT #
        015170  013746  015440                                                 MOV     RLUN,-(SP)
        015174  012746  015556                                                 MOV     #FM0,-(SP)
        015200  012746  000002                                                 MOV     #2,-(SP)
        015204  010600                                                         MOV     SP,R0
        015206  104416                                                         TRAP    C$PNTS
        015210  062706  000006                                                 ADD     #6,SP
   3153 015214                                PRINTS  #FM,#0,WRTN0(R5),RDN0(R5),<B,BDATA(R5)>,R1,R2,R3,R4
        015214  010446                                                         MOV     R4,-(SP)
        015216  010346                                                         MOV     R3,-(SP)
        015220  010246                                                         MOV     R2,-(SP)
        015222  010146                                                         MOV     R1,-(SP)
        015224  005046                                                         CLR     -(SP)
        015226  156516  000134                                                 BISB    BDATA(R5),(SP)
        015232  016546  000114                                                 MOV     RDN0(R5),-(SP)
        015236  016546  000110                                                 MOV     WRTN0(R5),-(SP)
        015242  012746  000000                                                 MOV     #0,-(SP)
        015246  012746  015574                                                 MOV     #FM,-(SP)
        015252  012746  000011                                                 MOV     #11,-(SP)
        015256  010600                                                         MOV     SP,R0
        015260  104416                                                         TRAP    C$PNTS
        015262  062706  000024                                                 ADD     #24,SP
   3154 015266  116501  000123                MOVB    SOFTR+1(R5),R1  ;SAME
   3155 015272  042701  177400                BIC     #177400,R1      ;AS
   3156 015276  116502  000125                MOVB    SOFTW+1(R5),R2  ;ABOVE
   3157 015302  042702  177400                BIC     #177400,R2      ;THIS
   3158 015306  116503  000137                MOVB    HARDR+1(R5),R3  ;TIME
   3159 015312  042703  177400                BIC     #177400,R3      ;FOR
   3160 015316  116504  000141                MOVB    HARDW+1(R5),R4  ;DRIVE
   3161 015322  042704  177400                BIC     #177400,R4      ;ONE
   3162
   3163 015326                                PRINTS  #FM,#1,WRTN1(R5),RDN1(R5),<B,BDATA+1(R5)>,R1,R2,R3,R4
        015326  010446                                                         MOV     R4,-(SP)
        015330  010346                                                         MOV     R3,-(SP)
        015332  010246                                                         MOV     R2,-(SP)
        015334  010146                                                         MOV     R1,-(SP)
        015336  005046                                                         CLR     -(SP)
        015340  156516  000135                                                 BISB    BDATA+1(R5),(SP)
        015344  016546  000116                                                 MOV     RDN1(R5),-(SP)
        015350  016546  000112                                                 MOV     WRTN1(R5),-(SP)
        015354  012746  000001                                                 MOV     #1,-(SP)
        015360  012746  015574                                                 MOV     #FM,-(SP)
        015364  012746  000011                                                 MOV     #11,-(SP)
        015370  010600                                                         MOV     SP,R0
```

```
                015372  104416                                                          TRAP    C$PNTS
                015374  062706  000024                                                  ADD     #24,SP
3164            015400  023727  015442  003364  2$:     CMP     RPTR,#LSTDEV    ;ALL UNITS DONE?
3165            015406  103005                          BHIS    3$              ;YES
3166            015410  062737  000002  015442          ADD     #2,RPTR         ;NO-DO
3167
3168            015416  000137  015102                  JMP     1$              ;MORE UNITS
3169
3170            015422                          3$:     POP     R5
                015422  012605                                                  MOV     (SP)+,R5
3171            015424                                  POP     R4
                015424  012604                                                  MOV     (SP)+,R4
3172            015426                                  POP     R3
                015426  012603                                                  MOV     (SP)+,R3
3173            015430                                  POP     R2
                015430  012602                                                  MOV     (SP)+,R2
3174            015432                                  POP     R1
                015432  012601                                                  MOV     (SP)+,R1
3175            015434                                  POP     R0
                015434  012600                                                  MOV     (SP)+,R0
3176            015436                                  ENDRPT
                015436                                                                  L10006:
                015436  104425                                                          TRAP    C$RPT
3177            015440  000000                  RLUN:   .WORD
3178            015442  000000                  RPTR:   .WORD
3179
3180            015444      045     116     045 STATHD: .ASCII  /%N%A       DR BLKS WR  BLKS RD  BDPAK  /
3181            015512      104     103     110         .ASCIZ  @DCHK/RD DCHK/WR  DCHK/RD DCHK/WR%N@
3182                                                    .EVEN
3183            015556      045     101     125 FM0:    .ASCIZ  /%AUNIT %D1%N/
3184                                                    .EVEN
3185
3186            015574      045     101     040 FM:     .ASCII  /%A       %D1%A %D5%A.    %D5%A.     %D3%A.    /
3187            015650      045     104     063         .ASCIZ  /%D3%A.   %D3%A.      %D3%A.     %D3%A.%N/
3188                                                    .EVEN
3189            015720      045     101     040 STHD2:  .ASCII  /%A                                        /
3190            015765      122     105     103         .ASCIZ  /RECOV   RECOV   UNRECOV UNRECOV%N/
3191                                                    .EVEN
3192            016030                                  ENDMOD
```

```
                                .SBTTL  INITIALIZE SECTION

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

                                        BGNINIT
                                                                                L$INIT::

                                INIT:   NOP                     ;
                                        CLRB    STRT            ;FOR STATS CLEAR
                                        CLR     TEST8           ;***** CLR TST 8 FLAG
                                        READEF  #EF.START       ;START COMMAND?
                                                                                MOV     #EF.START,R0
                                                                                TRAP    C$REFG
                                        BNCOMPLETE INIT2        ;NO
                                                                                BCC     INIT2
                                        INC     STRT            ;YES, SET START FLAG
                                INIT2:  MOV     #BLKTBL,DEVPTR  ;SET ALL UNITS ABORTED:
                                        CLR     R4              ;UNIT NUMBER
                                1$:     MOV     @DEVPTR,R5      ;GET POINTER
                                        MOV     R4,@R5          ;INSERT UNIT #
                                        BIS     #BIT15,@R5        ;SET ABORTED
                                        BIS     #BIT11,@R5      ;SET UNIT NOT TESTED
                                        ASL     R4              ;*2 FOR LOOK-UP
                                        MOV     BUFTBL(R4),RCVBUF(R5)   ;SETUP POINTER TO UNIT'S BUFFER
                                        ASR     R4              ;CORRECT BACK TO UNIT #
                                        CMP     DEVPTR,#LSTDEV  ;LAST DEVICE DONE?
                                        BHIS    CHECK           ;YES
                                        ADD     #2,DEVPTR       ;NO-GET
                                        INC     R4              ;NEXT DEVICE AND
                                        BR      1$              ;SERVICE

                                CHECK:  CMP     #8.,L$UNIT      ;MAKE SURE NOT
                                        BHIS    GETHRD          ;TOO MANY UNITS
                                        ERRSF   101.,TOMANY             ;TOMANY-REQUEST ^C
                                                                                TRAP    C$ERSF
                                                                                .WORD   101
                                                                                .WORD   TOMANY
                                                                                .WORD   0
                                        DOCLN                   ;EXIT
                                                                                TRAP    C$DCLN

                                GETHRD: MOV     #BLKTBL,DEVPTR  ;INIT TABLE POINTER
                                        CLR     R4              ;CLEAR DEVICE COUNTER
                                1$:     MOV     @DEVPTR,R5      ;GET STATUS WORD
                                        MOV     R4,L$LUN        ;UNIT NUM. IN CASE ERROR
                                        GPHARD  R4,R2           ;GET HARD INFO
                                                                                MOV     R4,R0
                                                                                TRAP    C$GPHRD
                                                                                MOV     R0,R2
                                        BNCOMPLETE 3$
                                                                                BCC     3$
                                        BIC     #BIT11,@R5      ;UNIT IS TESTED!
                                        MOV     (R2)+,R3        ;R3=CSR
                                        MOV     (R2)+,TUVECT(R5)  ;GET VECTOR ADDRESS
```

```
3239 016224  112265  000061            MOVB    (R2)+,DR+1(R5)  ;SAVE UNIT SUMMARY
3240 016230  005202                    INC     R2              ;GET TO WORD BOUND
3241 016232  012237  016614            MOV     (R2)+,PDTFLG    ;AND GET PDT FLAG
3242 016236  052715  040000            BIS     #BIT14,@R5      ;SET SEND BREAK FLAG
3243 016242  032765  000400  000060    BIT     #BIT8,DR(R5)    ;DRIVE 0?
3244 016250  001011                    BNE     13$             ;YES
3245 016252  032765  001000  000060    BIT     #BIT9,DR(R5)    ;DRIVE 1?
3246 016260  001005                    BNE     13$             ;OK
3247 016262                            ERRSF   102.,NODRVS             ;NEITHER?!
     016262  104454                                                    TRAP    C$ERSF
     016264  000146                                                    .WORD   102
     016266  016560                                                    .WORD   NODRVS
     016270  000000                                                    .WORD   0
3248 016272                            DOCLN                   ;EXIT
     016272  104444                                                    TRAP    C$DCLN
3249
3250 016274  105737  016612    13$:    TSTB    STRT            ;START COMMAND?
3251 016300  001412                    BEQ     14$             ;NO, DONT CLEAR
3252                                                           ;YES-CLEAR STATS
3253 016302  012702  000202            MOV     #BLKEND,R2      ;R2-->END OF STATS
3254 016306  012701  000110            MOV     #WRTNO,R1       ;FORM ADDRESS OF START:
3255 016312  060501                    ADD     R5,R1           ;R1-->START OF STATS.
3256 016314  162702  000110            SUB     #WRTNO,R2       ;FORM # TO CLEAR
3257
3258 016320  105021            2$:     CLRB    (R1)+           ;CLEAR 'EM
3259 016322  005302                    DEC     R2              ;MORE?
3260 016324  001375                    BNE     2$              ;YES
3261 016326  042715  100000    14$:    BIC     #BIT15,@R5        ;SET NOT ABORTED
3262 016332  010365  000022            MOV     R3,RCSR(R5)     ;GET DEVICE REGISTERS:
3263 016336  062703  000002            ADD     #2,R3
3264 016342  010365  000024            MOV     R3,RCDB(R5)
3265 016346  062703  000002            ADD     #2,R3
3266 016352  010365  000026            MOV     R3,XMSR(R5)
3267 016356  062703  000002            ADD     #2,R3
3268 016362  105737  016614            TSTB    PDTFLG          ;UNIT A PDT?
3269 016366  001402                    BEQ     4$              ;NO
3270 016370  162703  000004            SUB     #4,R3           ;YES...RCDB=XMDB
3271 016374  010365  000030    4$:     MOV     R3,XMDB(R5)
3272 016400  005065  000072            CLR     PATTEN(R5)      ;ZERO DATA PATTERN
3273 016404  005065  000002            CLR     RETRY(R5)       ;NO RETRIES
3274 016410  005065  000064            CLR     REC(R5)         ;NO RECORD
3275 016414  005065  000076            CLR     SUCCS(R5)       ;NO SUCCESS
3276 016420  005065  000074            CLR     DLV(R5)         ;NO DLV ERROR
3277 016424  005065  000210            CLR     MRSP(R5)        ;***** CLR MRSP INDICATOR
3278 016430  005037  003336            CLR     ALLGON          ;OK TO PRINT STATISTICS
3279 016434  062737  000002  003310    3$:     ADD     #2,DEVPTR       ;-->NEXT DEVICE
3280 016442  005204                    INC     R4              ;INCREMENT UNIT NUMBER
3281 016444  020437  002012            CMP     R4,L$UNIT       ;MORE UNITS?
3282 016450  001250                    BNE     1$              ;YES, GP HARD THE NEXT
3283
3284 016452  005037  003304            CLR     SYSTAT          ;SYSTEM STATUS WORD
3285 016456                            RFLAGS  FLGLOC          ;GET USER FLAGS
     016456  104421                                                    TRAP    C$RFLA
     016460  010037  016616                                            MOV     R0,FLGLOC
3286 016464  005037  003330    5$:     CLR     BLKER           ;NO ERROR
3287 016470  013737  002206  003306  SETLEN: MOV     LENGTH,TAPLEN   ;GET # OF RECORDS
3288 016476  006237  003306            ASR     TAPLEN          ;GET # BLOCKS PER TRACK
```

```
 3289 016502  012737  000200  003332          MOV     #200,SECREC     ;PRESET SECOND START AT 200
 3290 016510  022737  000200  003306          CMP     #200,TAPLEN     ;# BLKS > 128.?
 3291 016516  101003                          BHI     3$              ;NO-SWITCH TRACKS 2ND PASS
 3292 016520  012737  000400  003332          MOV     #400,SECREC     ;YES-START AT 400
 3302
 3314
 3315 016526                          3$:     ENDINIT
      016526                                                                  L10007:
      016526  104411                                                                  TRAP    C$INIT
 3316
 3317
 3318 016530     124     117     117  TOMANY: .ASCIZ  /TOO MANY UNITS MAX.=8 /
 3319                                          .EVEN
 3320 016560     123     105     114  NODRVS: .ASCIZ  /SELECT AT LEAST 1 DRIVE /
 3321                                          .EVEN
 3322 016612  000000                  STRT::  .WORD
 3323 016614  000000                  PDTFLG::.WORD           ;TU58 IS IN PDT
 3324 016616  000000                  FLGLOC::.WORD           ;USER FLAGS
```

```
                                        ;++
                                        ; THE AUTO DROP CODE IS INVOKED WHEN THE ADR FLAG IS SET AND CHECKS FOR
                                        ; A VALID INTERFACE LOCATION.  DROPS UNIT IF INTERFACE IS NOT THERE.
                                        ;--

                                        BGNAUTO
016620                                                                                          L$AUTO::
016620  000240                                  NOP             ;AUTO DROP ROUTINE
016622                                        SETVEC  #4,#TRPHND,#PRI07       ;GET BUS TRAP VEC.
016622  012746  000340                                                                        MOV     #PRI07,-(SP)
016626  012746  016730                                                                        MOV     #TRPHND,-(SP)
016632  012746  000004                                                                        MOV     #4,-(SP)
016636  012746  000003                                                                        MOV     #3,-(SP)
016642  104437                                                                              TRAP    C$SVEC
016644  062706  000010                                                                      ADD     #10,SP
016650  012737  003346  016726                MOV     #BLKTBL,TRPPTR  ;GET TOP OF DATA BLOCK TABLE
016656  017705  000044          1$:           MOV     @TRPPTR,R5      ;GET DATA BLOCK
016662  032715  104000                        BIT     #BIT15!BIT11,@R5  ;NOT TESTED OR ABORTED?
016666  100403                                BMI     2$              ;YES
016670  005775  000022                        TST     @RCSR(R5)       ;NO-VALID ADDRESS?
016674  000240                                NOP                     ;YES...(TRAP IF NOT)
016676  023727  016726  003364    2$:         CMP     TRPPTR,#LSTDEV  ;MORE TO TRY?
016704  103004                                BHIS    3$              ;NO
016706  062737  000002  016726                ADD     #2,TRPPTR       ;ON TO NEXT
016714  000760                                BR      1$              ;GET IT
016716                                3$:     CLRVEC  #4              ;RESTORE
016716  012700  000004                                                                        MOV     #4,R0
016722  104436                                                                              TRAP    C$CVEC
016724                                        ENDAUTO
016724                                                                                  L10010:
016724  104461                                                                              TRAP    C$AUTO
016726  000000                                TRPPTR: .WORD

                                        ;ILLEGAL ADDRESS TRAP HANDLER:

016730                                        TRPHND: PRINTF  #MSAUTO         ;SAY ''AUTO DROPPED''
016730  012746  016762                                                                        MOV     #MSAUTO,-(SP)
016734  012746  000001                                                                        MOV     #1,-(SP)
016740  010600                                                                              MOV     SP,R0
016742  104417                                                                              TRAP    C$PNTF
016744  062706  000004                                                                      ADD     #4,SP
016750  011500                                      MOV     @R5,R0          ;GET UNIT #
016752  042700  177770                            BIC     #177770,R0      ;MASK IT OFF
016756                                            DODU    R0              ;DROP HIM
016756  104451                                                                              TRAP    C$DODU
016760  000002                                    RTI
016762     045     101     101  MSAUTO: .ASCIZ  /%AAUTO DROP: %N/
```

```
                                .SBTTL  CLEANUP CODING SECTION

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

017002                                  BGNCLN
017002                                                                  L$CLEAN::
017002  005737  003336                  TST     ALLGON          ;ENTRANCE FROM ALL-UNITS-ABORTED?
017006  001004                          BNE     1$              ;YES-EXIT
017010  005737  002210                  TST     STAEOP          ;NO-STATS AT EOP?
017014  001401                          BEQ     1$              ;NO
017016                                  DORPT                   ;YES
017016  104424                                                          TRAP    C$DRPT

                                1$:     ENDCLN
017020                                                                  L10011:
017020  104412                                                          TRAP    C$CLEAN
```

```
                                .SBTTL  DROP UNIT SECTION

                                ;++
                                ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
                                ; TO NO LONGER BE TESTED.
                                ;--

017022                                  BGNDU
017022                                                                          L$DU::
                                                        ;R0=UNIT NUMBER
017022                                  PUSH    R0      ;SAVE IT
017022  010046                                          MOV     R0,-(SP)


017024                                  PUSH     R5             ;SAVE PRESENT UNIT POINTER
017024  010546                                          MOV     R5,-(SP)


017026  004737  017066                  CALL     GETR5          ;GET POINTER TO UNIT
017032  052715  100000                  BIS     #BIT15,@R5      ;SET ABORTED
017036                                  POP     R5              ;RESTORE PRESENT UNIT POINTER
017036  012605                                          MOV     (SP)+,R5

017040                                  POP     R0              ;RETRIEVE UNIT NUMBER
017040  012600                                          MOV     (SP)+,R0

017042                                  PRINTF  #ABOMSG,R0
017042  010046                                                                  MOV     R0,-(SP)
017044  012746  017120                                                          MOV     #ABOMSG,-(SP)
017050  012746  000002                                                          MOV     #2,-(SP)
017054  010600                                                                  MOV     SP,R0
017056  104417                                                                  TRAP    C$PNTF
017060  062706  000006                                                          ADD     #6,SP

                                ENDDU
017064                                                                  L10012:
017064  104453                                                                  TRAP    C$DU
017066  012737  003346  017116  GETR5:  MOV     #BLKTBL,PTR     ;-->UNIT 0
017074  017705  000016          1$:     MOV     @PTR,R5         ;GET STATUS WORD
017100  005300                          DEC     R0              ;CORRECT UNIT?
017102  100404                          BMI     2$              ;YES
017104  062737  000002  017116          ADD     #2,PTR          ;NO,-->NEXT
017112  000770                          BR      1$              ;CONTINUE
017114  000207                  2$:     RETURN
017116  000000                  PTR:    .WORD

017120      045     101     104 ABOMSG: .ASCIZ  /%ADROPPED UNIT %D1%N/
                                        .EVEN
```

```
                                .SBTTL  ADD UNIT SECTION

                                ;++
                                ; THE ADD-UNIT SECTION CONTAINS ANY CODE THE PROGRAMMER WISHES
                                ;  TO BE EXECUTED IN CONJUNCTION WITH THE ADDING OF A UNIT BACK
                                ;  TO THE TEST CYCLE.
                                ;--

017146                                  BGNAU
017146                                                                  L$AU::

                                ;THE INIT CODE CONTAINS ALL CODE NECESSARY TO ADD A UNIT.

                                        ENDAU
017146                                                                  L10013:
017146  104452                                                                  TRAP    C$AU
```

```
                                .SBTTL  TEST 1 / DEVICE SELF-DIAGNOSTIC EXECUTION

017150                                  BGNMOD
                                .NLIST  ME,BEX

017150                                  BGNTST
017150                                                                  T1::
017150                                  TSTID   #TST1
017150  012737  017214  003324                        MOV     #TST1,TSTTOP    ;SAVE ADDR OF TEST
017156  004737  006002                                CALL    SETUP           ;INIT UNITS TSTPC
017162  004737  005630                                CALL    SETDR           ;GET 1ST DRVS.
017166  004737  006050                                CALL    RUN             ;DO TEST
017172  004737  005526                                CALL    SWAPDR          ;GET NEXT DRVS.
017176  103004                                        BCC     64$             ;BR NO 2ND DRVS
017200  004737  006002                                CALL    SETUP           ;REINIT UNITS TSTPC
017204  004737  006050                                CALL    RUN             ;REPEAT TEST
017210                                          64$:                        ;DONE
017210                                  EXIT TST
017210  104432                                                  TRAP    C$EXIT
017212  000136                                                  .WORD   L10014-.

017214                                  TST1:   TUSELF

017214  012700  026174                        64$:    MOV     #TRBUF,R0       ;FORM COMMAND PACKET
017220  112710  000002                                MOVB    #RSCMND,@R0     ;COMMAND FLAG
017224  112760  000012  000001                        MOVB    #RSMSIZ,1(R0)   ;SIZE OF MESSAGE
017232  112760  000007  000002                        MOVB    #RSSSLF,2(R0)   ;SELF TEST OPERATION
017240  105060  000003                                CLRB    3(R0)           ;NO MODIFIER.
017244  005060  000004                                CLR     4(R0)           ;NO DRIVE OR SWITCHES
017250  005060  000006                                CLR     6(R0)           ;NO SEQUENCE NUMBER
017254  005060  000010                                CLR     8.(R0)          ;NO BYTES
017260  005060  000012                                CLR     10.(R0)         ;NO RECORD #
017264  012701  000012                                MOV     #RSMSIZ,R1      ;GET SIZE
017270  005721                                        TST     (R1)+           ;+2 FOR CHECKSUM
017272  012765  000016  000070                        MOV     #RSSNSZ,SNDCNT(R5)  ;SIZE TO SEND
017300  004737  013520                                CALL    CHKSUM          ;FORM CHECKSUM
017304  010110                                        MOV     R1,(R0)         ;INSERT INTO PACKET
017306  012765  000002  000034                        MOV     #RSEND,XSFLG(R5)  ;EXPECT END,
017314  012765  000016  000036                        MOV     #RSNDSZ,XSCNT(R5)  ;THIS BIG
017322  012765  000001  000032                        MOV     #1,XSPKNM(R5)   ;AND 1 PACKET
                                                                            ;SEND
017330  004737  006556                                CALL    RSVP            ;RETURN TO SCHEDULER
017334  032715  000010                                BIT     #BIT3,@R5       ;RETRY?(BAD FLAG)
017340  001325                                        BNE     64$             ;YES

017342  005237  003320                INC     DONE
017346  000207                        RETURN


017350                                  ENDTST
017350                                                          L10014:
017350  104401                                                          TRAP    C$ETST
```

```
                                .SBTTL  TEST 2 / SEEK EOT,BOT

                                        BGNTST
                                                                        T2::
                                        TSTID   #TST2
017352  012737  017416  003324                        MOV     #TST2,TSTTOP    ;SAVE ADDR OF TEST
017360  004737  006002                                CALL    SETUP           ;INIT UNITS TSTPC
017364  004737  005630                                CALL    SETDR           ;GET 1ST DRVS.
017370  004737  006050                                CALL    RUN             ;DO TEST
017374  004737  005526                                CALL    SWAPDR          ;GET NEXT DRVS.
017400  103004                                        BCC     64$             ;BR NO 2ND DRVS
017402  004737  006002                                CALL    SETUP           ;REINIT UNITS TSTPC
017406  004737  006050                                CALL    RUN             ;REPEAT TEST
017412                                          64$:                          ;DONE
                                        EXIT TST
017412  104432                                                        TRAP    C$EXIT
017414  000206                                                        .WORD   L10015-.

017416  005004                  TST2:   CLR     R4                  ;R4=INDEX INTO RECORD TABLE
017420  016465  017606  000064  1$:     MOV     RECDAT(R4),REC(R5)  ;GET THE RECORD

                                        TUSEEK  REC(R5),DR(R5)  ;SEEK IT

017426  012700  026174                          64$:    MOV     #TRBUF,R0       ;-->(POINT TO) XMIT BUFFER
017432  112710  000002                                  MOVB    #RSCMND,@R0     ;FORM COMMAND MESSAGE PACK
017436  112760  000012  000001                         MOVB    #RSMSIZ,1(R0)   ;THIS BIG
017444  112760  000005  000002                         MOVB    #RSSSEK,2(R0)   ;OP CODE IS SEEK
017452  016560  000064  000012                         MOV     REC(R5),10.(R0) ;TO THIS RECORD
017460  116560  000060  000004                         MOVB    DR(R5),4.(R0)   ;AND WHICH DRIVE
017466  105060  000003                                  CLRB    3.(R0)          ;NO MODIFIER
017472  105060  000005                                  CLRB    5.(R0)          ;NO SWITCHES
017476  005060  000006                                  CLR     6.(R0)          ;NO SEQUENCE #
017502  005060  000010                                  CLR     8.(R0)          ;NO BYTE COUNT
017506  012701  000012                                  MOV     #RSMSIZ,R1      ;GET COUNT
017512  005721                                         TST     (R1)+           ;PLUS FLAG + BCNT
                                                                                ;FOR CHECKSUM CALC
017514  004737  013520                                 CALL    CHKSUM          ;R0-->TOP   R1=# OF BYTES
017520  010110                                         MOV     R1,(R0)         ;INSERT INTO PACKET
                                                                                ;SET UP EXPECTATIONS:
017522  012765  000016  000070                         MOV     #RSSNSZ,SNDCNT(R5)  ;HOW MANY TO SEND
017530  112765  000002  000034                         MOVB    #RSCMND,XSFLG(R5)  ;EXPECT END PACK
017536  012765  000016  000036                         MOV     #RSNDSZ,XSCNT(R5)  ;COUNT WITH THIS
017544  012765  000001  000032                         MOV     #1.,XSPKNM(R5)  ;EXPECT ONLY 1 PACKET

017552  004737  006556                                 CALL    RSVP            ;SEND
                                                                                ;AND RETURN TO SCHEDULER
017556  032715  000010                                 BIT     #BIT3,@R5       ;RETRY (FLAG BYTE ERROR)?
017562  001321                                         BNE     64$             ;YES

017564  062704  000002                  ADD     #2,R4               ;POINT TO NEXT RECORD
017570  026427  017606  177777         CMP     RECDAT(R4),#-1.     ;LAST ONE DONE?
017576  001310                         BNE     1$                  ;NO-LOOP
017600  005237  003320                 INC     DONE                ;YES-SET DONE FLAG
017604  000207                         RETURN
```

```
3573 017606  000000                RECDAT: 0.      ;BOT
3574 017610  000200                        200     ;BOT OTHER TRACK
3575 017612  000177                        177     ;EOT
3576 017614  000377                        377     ;EOT OTHER TRACK
3577 017616  000400                        400     ;BOT AGAIN
3578 017620  177777                        -1.
3579 017622                                ENDTST
     017622                                                  L10015:
     017622  104401                                                  TRAP    C$ETST
```

```
                                .SBTTL  TEST 3 / HIGH ACTIVITY WRITE/READ

                                ; WRITE THEN READ VARYING DATA FOR ALL PHYSICALLY ADJACENT BLOCKS AROUND
                                ; A RECORD, GO HALF-WAY INTO REMAINING TAPE REPEAT UNTIL EOT.

                                        BGNTST
                                                                                        T3::
                                        TSTID   #TST3
017624  012737  017670  003324                        MOV     #TST3,TSTTOP    ;SAVE ADDR OF TEST
017632  004737  006002                                CALL    SETUP           ;INIT UNITS TSTPC
017636  004737  005630                                CALL    SETDR           ;GET 1ST DRVS.
017642  004737  006050                                CALL    RUN             ;DO TEST
017646  004737  005526                                CALL    SWAPDR          ;GET NEXT DRVS.
017652  103004                                        BCC     64$             ;BR NO 2ND DRVS
017654  004737  006002                                CALL    SETUP           ;REINIT UNITS TSTPC
017660  004737  006050                                CALL    RUN             ;REPEAT TEST
017664                                          64$:                            ;DONE
                                        EXIT TST
017664  104432                                                                  TRAP    C$EXIT
017666  001340                                                                  .WORD   L10016-.

017670  012765  000100  000066  TST3:   MOV     #100,TMP(R5)    ;INIT TO HALF OF REMAINING
017676  005004                  CLR     R4              ;FOR INDEX INTO DATA TABLE
017700  005065  000064          CLR     REC(R5)         ;START AT RECORD 0
017704  016465  021214  000072  1$:     MOV     TST3PT(R4),PATTEN(R5)   ;GET DATA
                                TUWRIT  PATTEN(R5),REC(R5),#512.,DR(R5),#0
017712  012700  026174                          72$:    MOV     #TRBUF,R0       ;MAKE COMMAND PACKET:
017716  112710  000002                                  MOVB    #RSCMND,@R0     ;COMMAND FLAG
017722  112760  000012  000001                          MOVB    #RSMSIZ,1(R0)   ;THIS SIZE
017730  112760  000003  000002                          MOVB    #RSSWR,2(R0)    ;INSERT OP CODE-WRITE
017736  112760  000000  000003                          MOVB    #0,3.(R0)       ;VERIFY (1 OR 0)
017744  116560  000060  000004                          MOVB    DR(R5),4.(R0)   ;DRIVE #
017752  112760  000020  000005                          MOVB    #020,5.(R0)     ;MAINTENANCE MODE SWITCH
017760  005060  000006                                  CLR     6.(R0)          ;NO SEQUENCE #
017764  012760  001000  000010                          MOV     #512.,8.(R0)    ;TOTAL COUNT TO WRITE
017772  016560  000064  000012                          MOV     REC(R5),10.(R0) ;AT  RECORD N
020000  012701  000012                                  MOV     #RSMSIZ,R1      ;THE PACKET SIZE PLUS+2
020004  005721                                          TST     (R1)+           ;(FLAG AND COUNT) INTO R1
020006  012765  000016  000070                          MOV     #RSSNSZ,SNDCNT(R5)  ;LOAD THE  SIZE TO SEND
020014  004737  013520                                  CALL    CHKSUM          ;R0 --> R1=COUNT
020020  010110                                          MOV     R1,(R0)         ;PUT CHKSUM IN PACKET
                                                                                ;SET UP EXPECTATIONS:
020022  012765  000020  000034                          MOV     #RSCONT,XSFLG(R5)   ;THE FLAG
020030  012765  000001  000036                          MOV     #1,XSCNT(R5)    ;THE COUNT
020036  012765  000001  000032                          MOV     #1,XSPKNM(R5)   ;THE # PACKETS EXPECTED
020044  012702  001000                                  MOV     #512.,R2                ;GET # OF DATA BYTES
020050  004737  006556                                  CALL    RSVP            ;SEND (AND RETURN TO SCHEDULER)
020054  032715  000010                                  BIT     #BIT3,@R5       ;FLAG BYTE ERROR?
020060  001314                                          BNE     72$             ;YES
020062  042715  010000                                  BIC     #BIT12,@R5      ;FLAG FOR LAST PACKET
020066  012700  026174                          64$:    MOV     #TRBUF,R0       ;POINT TO TOP OF BUFFER AGAIN
020072  020227  000200                                  CMP     R2,#128.        ;START DATA PACKET(S)
020076  101004                                          BHI     65$             ;#512. > 128.!
020100  010201                                          MOV     R2,R1           ;#512.<128.
020102  052715  010000                                  BIS     #BIT12,@R5      ;SO LAST PACKET NOW
020106  000402                                          BR      66$             ;USE REMAINING COUNT
```

```
020110  012701  000200                65$:    MOV     #128.,R1        ;USE 128. BYTES
020114  110160  000001                66$:    MOVB    R1,1(R0)        ;COPY COUNT TO BUFFER
020120  010103                            MOV     R1,R3           ;R3=COUNTER TO LOAD BUFFER
020122  112710  000001                    MOVB    #RSDATA,@R0     ;FLAG FIRST
020126  005720                            TST     (R0)+           ;SKIP COUNT
020130  116520  000072                67$:    MOVB    PATTEN(R5),(R0)+        ;INSERT DATA
020134  005303                            DEC     R3              ;MORE?
020136  101374                            BHI     67$             ;YES
020140  012700  026174                    MOV     #TRBUF,R0       ;-->TOP AGAIN
020144  116001  000001                    MOVB    1(R0),R1        ;GET COUNT
020150  042701  177400                    BIC     #177400,R1      ;ZERO SIGN EXTEND
020154  010165  000070                    MOV     R1,SNDCNT(R5)   ;HOW MANY TO SEND PLUS
020160  062765  000004  000070            ADD     #4,SNDCNT(R5)   ;FLAG,COUNT,CHKSUM
020166  062701  000002                    ADD     #2,R1           ;COMPENSATE FOR FLAG + COUNT
020172  004737  013520                    CALL    CHKSUM          ;FOR CHECKSUM CALC.
020176  110120                            MOVB    R1,(R0)+        ;CHKSUM INTO PACKET
020200  000301                            SWAB    R1              ;EVEN ON AN ODD
020202  110120                            MOVB    R1,(R0)+        ;BYTE BOUNDARY
020204  032715  010000                    BIT     #BIT12,@R5      ;LAST DATA PACKET?
020210  001412                            BEQ     68$             ;NO
020212  012765  000002  000034            MOV     #RSEND,XSFLG(R5)  ;YES-EXPECT 'END'
020220  012765  000016  000036            MOV     #RSNDSZ,XSCNT(R5)  ;OF THIS SIZE
020226  012765  000001  000032            MOV     #1,XSPKNM(R5)   ;AND 1 PACKET
020234  000411                            BR      69$             ;SEND
020236  012765  000020  000034    68$:    MOV     #RSCONT,XSFLG(R5)  ;(NOT LAST), EXPECT 'CONTINUE'
020244  012765  000001  000036            MOV     #1,XSCNT(R5)    ;AND 1 BYTE
020252  012765  000001  000032            MOV     #1,XSPKNM(R5)   ;AND 1 PACKET
020260  004737  006556            69$:    CALL    RSVP            ;SEND PACKET
                                                                ;AND RETURN TO SCHEDULER
020264  032715  000010                    BIT     #BIT3,@R5       ;FLAG BYTE RETRY?
020270  001210                            BNE     72$             ;YES
020272  032715  002000                    BIT     #BIT10,@R5      ;RETRY DATA ERROR?
020276  001004                            BNE     70$             ;YES
020300  162702  000200                    SUB     #128.,R2        ;NO, MORE DATA TO SEND?
020304  101270                            BHI     64$             ;YES
020306  000502                            BR      71$             ;NO
020310                            70$:    TURTRY  REC(R5),#512.,DR(R5)    ;RETRY HERE


020310  012700  026174            76$:    MOV     #TRBUF,R0       ;FORM CMND PACK:
020314  112710  000002                    MOVB    #RSCMND,@R0     ;MESSAGE PACK TYPE
020320  112760  000012  000001            MOVB    #RSMSIZ,1(R0)   ;THIS BIG
020326  112760  000002  000002            MOVB    #RSSRD,2(R0)    ;OP CODE-READ
020334  016560  000064  000012            MOV     REC(R5),10.(R0) ;THIS RECORD
020342  116560  000060  000004            MOVB    DR(R5),4.(R0)   ;THIS DRIVE
020350  105060  000003                    CLRB    3(R0)           ;PRESET NORM THRESHOLD
020354  105715                            TSTB    @R5             ;REDUCED?
020356  100002                            BPL     77$             ;NO
020360  105260  000003                    INCB    3(R0)           ;YES-CHANGE THRESHOLD
020364  012760  001000  000010    77$:    MOV     #512.,8.(R0)    ;# BYTES DESIRED
020372  112760  000020  000005            MOVB    #020,5.(R0)     ;MAINTENANCE MODE
020400  005060  000006                    CLR     6.(R0)          ;NO SEQUENCE #
020404  012701  000012                    MOV     #RSMSIZ,R1      ;SIZE OF PACKET
020410  005721                            TST     (R1)+           ;PLUS FLAG+COUNT INTO R1
020412  012765  000016  000070            MOV     #RSSNSZ,SNDCNT(R5)  ;SET UP SIZE TO SEND

020420  004737  013520                    CALL    CHKSUM          ;FORM CHECKSUM R1=COUNT
```

```
        020424  010110                                          MOV     R1,(R0)         ;INSERT IN PACKET

        020426  012701  001000                                  MOV     #512.,R1                ;SET EXPECTATIONS:
                                                                                        ;CALC # OF DATA PACKETS TO EXPECT
        020432  012703  000034                                  MOV     #XSFLG,R3       ;OFFSET OF FLAG
        020436  060503                                          ADD     R5,R3           ;ABS. ADDR. OF XSFLG
        020440  005002                                          CLR     R2              ;PRESET
        020442  005202                                  73$:    INC     R2              ;# PACKETS EXPECTED
        020444  012723  000001                                  MOV     #RSDATA,(R3)+   ;LOAD XSFLG
        020450  012723  000204                                  MOV     #132.,(R3)+     ;AND EXPECT COUNT
        020454  162701  000200                                  SUB     #128.,R1        ;NEG RESULT LAST TIME
        020460  101401                                          BLOS    75$             ;LAST TIME!
        020462  000767                                          BR      73$             ;MORE TO DO
        020464  005202                                  75$:    INC     R2              ;ADD ONE FOR END PACK
        020466  010265  000032                                  MOV     R2,XSPKNM(R5)   ;SAVE # PACKETS TO EXPECT
        020472  012723  000002                                  MOV     #RSEND,(R3)+    ;EXPECT AN END
        020476  012713  000016                                  MOV     #RSNDSZ,(R3)    ;THIS BIG-14. BYTES

        020502  004737  006556                                  CALL    RSVP            ;SEND
                                                                                        ;AND RETURN TO SCHEDULER

3597    020516                          TUREAD  REC(P5),#512.,DR(R5),#0


        020516  012700  026174                          82$:    MOV     #TRBUF,R0       ;FORM CMND PACK:
        020522  112710  000002                                  MOVB    #RSCMND,@R0     ;MESSAGE PACK TYPE
        020526  112760  000012  000001                         MOVB    #RSMSIZ,1(R0)   ;THIS BIG
        020534  112760  000002  000002                         MOVB    #RSSRD,2(R0)    ;OP CODE IS READ
        020542  016560  000064  000012                         MOV     REC(R5),10.(R0) ;THIS RECORD
        020550  116560  000060  000004                         MOVB    DR(R5),4.(R0)   ;THIS DRIVE
        020556  112760  000000  000003                         MOVB    #0,3.(R0)       ;VERIFY
        020564  012760  001000  000010                         MOV     #512.,8.(R0)    ;TOTAL BYTES TO READ
        020572  112760  000020  000005                         MOVB    #020,5.(R0)     ;MAINTENANCE MODE
        020600  005060  000006                                  CLR     6.(R0)          ;NO SEQUENCE #
        020604  012701  000012                                  MOV     #RSMSIZ,R1      ;GET SIZE OF PACKET
        020610  005721                                          TST     (R1)+           ;+2 FOR CHECKSUM
        020612  012765  000016  000070                         MOV     #RSSNSZ,SNDCNT(R5)  ;SIZE TO SEND
        020620  004737  013520                                  CALL    CHKSUM          ;FORM CHECKSUM R1=COUNT
        020624  010110                                          MOV     R1,(R0)         ;INSERT CHECKSUM

        020626  012701  001000                                  MOV     #512.,R1                ;SET EXPECTATIONS:
                                                                                        ;CALC # OF DATA PACKETS TO EXPECT:
        020632  012703  000034                                  MOV     #XSFLG,R3       ;GET OFFSET
        020636  060503                                          ADD     R5,R3           ;ABS. ADDR. OF XSFLG
        020640  005002                                          CLR     R2              ;PRESET AS NONE
        020642  005202                                  78$:    INC     R2              ;# PACKETS EXPECTED
        020644  012723  000001                                  MOV     #RSDATA,(R3)+   ;LOAD XSFLG
        020650  012723  000204                                  MOV     #132.,(R3)+     ;AND EXPECTED COUNT
        020654  162701  000200                                  SUB     #128.,R1        ;NEG RESULT LAST TIME
        020660  101401                                          BLOS    80$             ;LAST TIME
        020662  000767                                          BR      78$             ;MORE TO DO
        020664  005202                                  80$:    INC     R2              ;ADD ONE FOR END PACK
        020666  010265  000032                                  MOV     R2,XSPKNM(R5)   ;SAVE # PACKETS TO EXPECT
        020672  012723  000002                                  MOV     #RSEND,(R3)+    ;EXPECT AN END ALSO...
        020676  012713  000016                                  MOV     #RSNDSZ,(R3)    ;THIS BIG-14. BYTES
        020702  004737  006556                                  CALL    RSVP            ;SEND
                                                                                        ;AND RETURN TO SCHEDULER
```

```
020706  032715  002010                  81$:    BIT     #BIT10!BIT3,@R5 ;RETRY?
020712  001500                                  BEQ     79$             ;NO.
020714                                          TURTRY  REC(R5),#512.,DR(R5)    ;YES


020714  012700  026174                  86$:    MOV     #TRBUF,R0       ;FORM CMND PACK:
020720  112710  000002                          MOVB    #RSCMND,@R0     ;MESSAGE PACK TYPE
020724  112760  000012  000001                  MOVB    #RSMSIZ,1(R0)   ;THIS BIG
020732  112760  000002  000002                  MOVB    #RSSRD,2(R0)    ;OP CODE-READ
020740  016560  000064  000012                  MOV     REC(R5),10.(R0) ;THIS RECORD
020746  116560  000060  000004                  MOVB    DR(R5),4.(R0)   ;THIS DRIVE
020754  105060  000003                          CLRB    3(R0)           ;PRESET NORM THRESHOLD
020760  105715                                  TSTB    @R5             ;REDUCED?
020762  100002                                  BPL     87$             ;NO
020764  105260  000003                          INCB    3(R0)           ;YES-CHANGE THRESHOLD
020770  012760  001000  000010          87$:    MOV     #512.,8.(R0)    ;# BYTES DESIRED
020776  112760  000020  000005                  MOVB    #020,5.(R0)     ;MAINTENANCE MODE
021004  005060  000006                          CLR     6.(R0)          ;NO SEQUENCE #
021010  012701  000012                          MOV     #RSMSIZ,R1      ;SIZE OF PACKET
021014  005721                                  TST     (R1)+           ;PLUS FLAG+COUNT INTO R1
021016  012765  000016  000070                  MOV     #RSSNSZ,SNDCNT(R5)  ;SET UP SIZE TO SEND

021024  004737  013520                          CALL    CHKSUM          ;FORM CHECKSUM R1=COUNT
021030  010110                                  MOV     R1,(R0)         ;INSERT IN PACKET

021032  012701  001000                          MOV     #512.,R1                ;SET EXPECTATIONS:
                                                                        ;CALC # OF DATA PACKETS TO EXPECT
021036  012703  000034                          MOV     #XSFLG,R3       ;OFFSET OF FLAG
021042  060503                                  ADD     R5,R3           ;ABS. ADDR. OF XSFLG
021044  005002                                  CLR     R2              ;PRESET
021046  005202                          83$:    INC     R2              ;# PACKETS EXPECTED
021050  012723  000001                          MOV     #RSDATA,(R3)+   ;LOAD XSFLG
021054  012723  000204                          MOV     #132.,(R3)+     ;AND EXPECT COUNT
021060  162701  000200                          SUB     #128.,R1        ;NEG RESULT LAST TIME
021064  101401                                  BLOS    85$             ;LAST TIME!
021066  000767                                  BR      83$             ;MORE TO DO
021070  005202                          85$:    INC     R2              ;ADD ONE FOR END PACK
021072  010265  000032                          MOV     R2,XSPKNM(R5)   ;SAVE # PACKETS TO EXPECT
021076  012723  000002                          MOV     #RSEND,(R3)+    ;EXPECT AN END
021102  012713  000016                          MOV     #RSNDSZ,(R3)    ;THIS BIG-14. BYTES

021106  004737  006556                          CALL    RSVP            ;SEND
                                                                        ;AND RETURN TO SCHEDULER

3598 021116  062704  000002                  ADD     #2,R4           ;POINT TO NEXT DATA
3599 021122  005764  021214                  TST     TST3PT(R4)      ;END?
3600 021126  001402                          BEQ     2$              ;YES
3601 021130  000137  017704                  JMP     1$              ;NO-WRITE, READ NEW DATA
3602 021134  005004                  2$:     CLR     R4              ;POINT TO FIRST DATA
3603 021136  062765  000200  000064          ADD     #200,REC(R5)    ;BUT NOW USE ADJACENT RECORD
3604 021144  032765  001000  000064          BIT     #1000,REC(R5)   ;ALL ADJACENT RECORDS DONE?
3605 021152  001002                          BNE     3$              ;YES
3606 021154  000137  017704                  JMP     1$              ;NO-WRITE, READ AT NEW RECORD
3607 021160  162765  001000  000064  3$:     SUB     #1000,REC(R5)   ;RESTORE TO NEXT RECORD
3608 021166  066565  000066  000064          ADD     TMP(R5),REC(R5) ;HALF INTO REST OF TAPE
3609 021174  006265  000066                  ASR     TMP(R5)         ;HALF OF HALF FOR NEXT TIME
3610 021200  103402                          BCS     4$              ;DONE?
```

```
3611 021202  000137  017704                JMP     1$              ;NO
3612 021206  005237  003320        4$:     INC     DONE            ;YES-SET FLAG
3613 021212  000207                        RETURN
3614 021214  000000                TST3PT: .WORD   000000
3615 021216  125252                        .WORD   125252
3616 021220  177777                        .WORD   177777
3617 021222  052525                        .WORD   052525
3618 021224  000000                        .WORD   000000
3619
3620
3621 021226                                ENDTST
     021226                                                  L10016:
     021226  104401                                                  TRAP    C$ETST
```

```
3624
3625                                    .SBTTL  TEST 4 / WRITE SELECTED NUMBER OF BLOCKS
3626
3627 021230                                    BGNTST
     021230                                                                          T4::
3628 021230                                    TSTID   #TST4
     021230  012737  021274  003324                            MOV     #TST4,TSTTOP    ;SAVE ADDR OF TEST
     021236  004737  006002                            CALL    SETUP           ;INIT UNITS TSTPC
     021242  004737  005630                            CALL    SETDR           ;GET 1ST DRVS.
     021246  004737  006050                            CALL    RUN             ;DO TEST
     021252  004737  005526                            CALL    SWAPDR          ;GET NEXT DRVS.
     021256  103004                                    BCC     64$             ;BR NO 2ND DRVS
     021260  004737  006002                            CALL    SETUP           ;REINIT UNITS TSTPC
     021264  004737  006050                            CALL    RUN             ;REPEAT TEST
     021270                                    64$:                            ;DONE
3629 021270                                    EXIT TST
     021270  104432                                                                    TRAP    C$EXIT
     021272  000724                                                                    .WORD   L10017-.
3630
3631
3632 021274  005065  000064            TST4:   CLR     REC(R5)         ;START AT REC 0
3633 021300  013765  003306  000066            MOV     TAPLEN,TMP(R5)  ;GET THE # OF BLOCKS PER TRACK
3634 021306  005065  000062            CLR     TRK(R5)         ;TRK(R5)=1ST OR 2ND PASS COUNTER
3635 021312  016565  000064  000072    1$:     MOV     REC(R5),PATTEN(R5)  ;USE RECORD NO. FOR DATA
3636 021320  005737  002216            TST     DRVCHK                  ;ADD DR #?
3637 021324  001403                    BEQ     10$                     ;NO
3638 021326  066565  000060  000072    ADD     DR(R5),PATTEN(R5)       ;YES, ADD DRIVE ID
3639 021334                            10$:    TUWRIT  PATTEN(R5),REC(R5),#512.,DR(R5),#0
     021334  012700  026174                    72$:    MOV     #TRBUF,R0       ;MAKE COMMAND PACKET:
     021340  112710  000002                            MOVB    #RSCMND,@R0     ;COMMAND FLAG
     021344  112760  000012  000001                    MOVB    #RSMSIZ,1(R0)   ;THIS SIZE
     021352  112760  000003  000002                    MOVB    #RSSWR,2(R0)    ;INSERT OP CODE-WRITE
     021360  112760  000000  000003                    MOVB    #0,3.(R0)       ;VERIFY (1 OR 0)
     021366  116560  000060  000004                    MOVB    DR(R5),4.(R0)   ;DRIVE #
     021374  112760  000020  000005                    MOVB    #020,5.(R0)     ;MAINTENANCE MODE SWITCH
     021402  005060  000006                            CLR     6.(R0)          ;NO SEQUENCE #
     021406  012760  001000  000010                    MOV     #512.,8.(R0)    ;TOTAL COUNT TO WRITE
     021414  016560  000064  000012                    MOV     REC(R5),10.(R0) ;AT  RECORD N
     021422  012701  000012                            MOV     #RSMSIZ,R1      ;THE PACKET SIZE PLUS+2
     021426  005721                                    TST     (R1)+           ;(FLAG AND COUNT) INTO R1
     021430  012765  000016  000070                    MOV     #RSSNSZ,SNDCNT(R5)  ;LOAD THE  SIZE TO SEND
     021436  004737  013520                            CALL    CHKSUM          ;R0 --> R1=COUNT
     021442  010110                                    MOV     R1,(R0)         ;PUT CHKSUM IN PACKET
                                                                               ;SET UP EXPECTATIONS:
     021444  012765  000020  000034                    MOV     #RSCONT,XSFLG(R5)   ;THE FLAG
     021452  012765  000001  000036                    MOV     #1,XSCNT(R5)    ;THE COUNT
     021460  012765  000001  000032                    MOV     #1,XSPKNM(R5)   ;THE # PACKETS EXPECTED
     021466  012702  001000                            MOV     #512.,R2                ;GET # OF DATA BYTES
     021472  004737  006556                            CALL    RSVP            ;SEND (AND RETURN TO SCHEDULER)
     021476  032715  000010                            BIT     #BIT3,@R5       ;FLAG BYTE ERROR?
     021502  001314                                    BNE     72$             ;YES
     021504  042715  010000                            BIC     #BIT12,@R5      ;FLAG FOR LAST PACKET
     021510  012700  026174                    64$:    MOV     #TRBUF,R0       ;POINT TO TOP OF BUFFER AGAIN
     021514  020227  000200                            CMP     R2,#128.        ;START DATA PACKET(S)
     021520  101004                                    BHI     65$             ;#512. > 128.!
     021522  010201                                    MOV     R2,R1           ;#512.<128.
     021524  052715  010000                            BIS     #BIT12,@R5      ;SO LAST PACKET NOW
```

```
021530  000402                                          BR      66$             ;USE REMAINING COUNT
021532  012701  000200                        65$:     MOV     #128.,R1        ;USE 128. BYTES
021536  110160  000001                        66$:     MOVB    R1,1(R0)        ;COPY COUNT TO BUFFER
021542  010103                                          MOV     R1,R3           ;R3=COUNTER TO LOAD BUFFER
021544  112710  000001                                MOVB    #RSDATA,@R0     ;FLAG FIRST
021550  005720                                          TST     (R0)+           ;SKIP COUNT
021552  116520  000072                        67$:     MOVB    PATTEN(R5),(R0)+        ;INSERT DATA
021556  005303                                          DEC     R3              ;MORE?
021560  101374                                          BHI     67$             ;YES
021562  012700  026174                                MOV     #TRBUF,R0       ;-->TOP AGAIN
021566  116001  000001                                MOVB    1(R0),R1        ;GET COUNT
021572  042701  177400                                BIC     #177400,R1      ;ZERO SIGN EXTEND
021576  010165  000070                                MOV     R1,SNDCNT(R5)   ;HOW MANY TO SEND PLUS
021602  062765  000004  000070                       ADD     #4,SNDCNT(R5)   ;FLAG,COUNT,CHKSUM
021610  062701  000002                                ADD     #2,R1           ;COMPENSATE FOR FLAG + COUNT
021614  004737  013520                                CALL    CHKSUM          ;FOR CHECKSUM CALC.
021620  110120                                          MOVB    R1,(R0)+        ;CHKSUM INTO PACKET
021622  000301                                          SWAB    R1              ;EVEN ON AN ODD
021624  110120                                          MOVB    R1,(R0)+        ;BYTE BOUNDARY
021626  032715  010000                                BIT     #BIT12,@R5      ;LAST DATA PACKET?
021632  001412                                          BEQ     68$             ;NO
021634  012765  000002  000034                       MOV     #RSEND,XSFLG(R5)  ;YES-EXPECT 'END'
021642  012765  000016  000036                       MOV     #RSNDSZ,XSCNT(R5)  ;OF THIS SIZE
021650  012765  000001  000032                       MOV     #1,XSPKNM(R5)   ;AND 1 PACKET
021656  000411                                          BR      69$             ;SEND
021660  012765  000020  000034              68$:     MOV     #RSCONT,XSFLG(R5)  ;(NOT LAST), EXPECT 'CONTINUE'
021666  012765  000001  000036                       MOV     #1,XSCNT(R5)    ;AND 1 BYTE
021674  012765  000001  000032                       MOV     #1,XSPKNM(R5)   ;AND 1 PACKET
021702  004737  006556                        69$:     CALL    RSVP            ;SEND PACKET
                                                                                ;AND RETURN TO SCHEDULER
021706  032715  000010                                BIT     #BIT3,@R5       ;FLAG BYTE RETRY?
021712  001210                                          BNE     72$             ;YES
021714  032715  002000                                BIT     #BIT10,@R5      ;RETRY DATA ERROR?
021720  001004                                          BNE     70$             ;YES
021722  162702  000200                                SUB     #128.,R2        ;NO, MORE DATA TO SEND?
021726  101270                                          BHI     64$             ;YES
021730  000502                                          BR      71$             ;NO
021732                                          70$:     TURTRY  REC(R5),#512.,DR(R5)    ;RETRY HERE


021732  012700  026174                        76$:     MOV     #TRBUF,R0       ;FORM CMND PACK:
021736  112710  000002                                MOVB    #RSCMND,@R0     ;MESSAGE PACK TYPE
021742  112760  000012  000001                       MOVB    #RSMSIZ,1(R0)   ;THIS BIG
021750  112760  000002  000002                       MOVB    #RSSRD,2(R0)    ;OP CODE-READ
021756  016560  000064  000012                       MOV     REC(R5),10.(R0) ;THIS RECORD
021764  116560  000060  000004                       MOVB    DR(R5),4.(R0)   ;THIS DRIVE
021772  105060  000003                                CLRB    3(R0)           ;PRESET NORM THRESHOLD
021776  105715                                          TSTB    @R5             ;REDUCED?
022000  100002                                          BPL     77$             ;NO
022002  105260  000003                                INCB    3(R0)           ;YES-CHANGE THRESHOLD
022006  012760  001000  000010              77$:     MOV     #512.,8.(R0)    ;# BYTES DESIRED
022014  112760  000020  000005                       MOVB    #020,5.(R0)     ;MAINTENANCE MODE
022022  005060  000006                                CLR     6.(R0)          ;NO SEQUENCE #
022026  012701  000012                                MOV     #RSMSIZ,R1      ;SIZE OF PACKET
022032  005721                                          TST     (R1)+           ;PLUS FLAG+COUNT INTO R1
022034  012765  000016  000070                       MOV     #RSSNSZ,SNDCNT(R5)  ;SET UP SIZE TO SEND
```

```
        022042  004737  013520                          CALL    CHKSUM          ;FORM CHECKSUM R1=COUNT
        022046  010110                                  MOV     R1,(R0)         ;INSERT IN PACKET

        022050  012701  001000                          MOV     #512.,R1                ;SET EXPECTATIONS:
                                                                                ;CALC # OF DATA PACKETS TO EXPECT
        022054  012703  000034                          MOV     #XSFLG,R3       ;OFFSET OF FLAG
        022060  060503                                  ADD     R5,R3           ;ABS. ADDR. OF XSFLG
        022062  005002                                  CLR     R2              ;PRESET
        022064  005202                          73$:    INC     R2              ;# PACKETS EXPECTED
        022066  012723  000001                          MOV     #RSDATA,(R3)+   ;LOAD XSFLG
        022072  012723  000204                          MOV     #132.,(R3)+     ;AND EXPECT COUNT
        022076  162701  000200                          SUB     #128.,R1        ;NEG RESULT LAST TIME
        022102  101401                                  BLOS    75$             ;LAST TIME!
        022104  000767                                  BR      73$             ;MORE TO DO
        022106  005202                          75$:    INC     R2              ;ADD ONE FOR END PACK
        022110  010265  000032                          MOV     R2,XSPKNM(R5)   ;SAVE # PACKETS TO EXPECT
        022114  012723  000002                          MOV     #RSEND,(R3)+    ;EXPECT AN END
        022120  012713  000016                          MOV     #RSNDSZ,(R3)    ;THIS BIG-14. BYTES

        022124  004737  006556                          CALL    RSVP            ;SEND
                                                                                ;AND RETURN TO SCHEDULER

  3640  022140  005365  000066                  DEC     TMP(R5)         ;DO ALL RECORDS FOR THIS TRACK?
  3641  022144  001404                          BEQ     2$              ;YES-GET OTHER TRACK
  3642  022146  005265  000064                  INC     REC(R5)         ;NO-ONTO NEXT RECORD
  3643  022152  000137  021312                  JMP     1$              ;EXECUTE THE WRITE
  3644  022156  005765  000062          2$:     TST     TRK(R5)         ;DONE 2 TRACKS?
  3645  022162  001012                          BNE     TST4EX          ;YES-EXIT
  3646  022164  005265  000062                  INC     TRK(R5)         ;NO-SET FLAG FOR NEXT PASS
  3647  022170  013765  003332  000064          MOV     SECREC,REC(R5)  ;GET NEW STARTING BLOCK #
  3648  022176  013765  003306  000066          MOV     TAPLEN,TMP(R5)  ;RESET # OF BLOCKS
  3649  022204  000137  021312                  JMP     1$              ;AND EXECUTE
  3650  022210  005237  003320          TST4EX: INC     DONE            ;DONE
  3651  022214  000207                          RETURN                  ;RETURN
  3652
  3653  022216                                  ENDTST
        022216                                                                  L10017:
        022216  104401                                                                  TRAP    C$ETST
```

```
                                .SBTTL  TEST 5 / READ SELECTED NUMBER OF BLOCKS

                                        BGNTST
                                                                                T5::
                                        TSTID   #TST5
022220  012737  022264  003324                        MOV     #TST5,TSTTOP    ;SAVE ADDR OF TEST
022226  004737  006002                                CALL    SETUP           ;INIT UNITS TSTPC
022232  004737  005630                                CALL    SETDR           ;GET 1ST DRVS.
022236  004737  006050                                CALL    RUN             ;DO TEST
022242  004737  005526                                CALL    SWAPDR          ;GET NEXT DRVS.
022246  103004                                        BCC     64$             ;BR NO 2ND DRVS
022250  004737  006002                                CALL    SETUP           ;REINIT UNITS TSTPC
022254  004737  006050                                CALL    RUN             ;REPEAT TEST
022260                                          64$:                          ;DONE
                                        EXIT TST
022260  104432                                                        TRAP    C$EXIT
022262  000520                                                        .WORD   L10020-.

022264  005065  000064          TST5:   CLR     REC(R5)                 ;START AT REC 0
022270  013765  003306  000066          MOV     TAPLEN,TMP(R5)          ;GET THE # OF BLOCKS PER TRACK
022276  005065  000062                  CLR     TRK(R5)                 ;TRK(R5)=1ST OR 2ND PASS
022302  016565  000064  000072  1$:     MOV     REC(R5),PATTEN(R5)      ;USE RECORD NO. AS DATA
022310  005737  002216                  TST     DRVCHK                  ;ADD DR #?
022314  001403                          BEQ     10$                     ;NO
022316  066565  000060  000072          ADD     DR(R5),PATTEN(R5)       ;ADD IN DRIVE ID
022324                          10$:    TUREAD  REC(R5),#512.,DR(R5),#0


022324  012700  026174                                  68$:    MOV     #TRBUF,R0       ;FORM CMND PACK:
022330  112710  000002                                  MOVB    #RSCMND,@R0     ;MESSAGE PACK TYPE
022334  112760  000012  000001                          MOVB    #RSMSIZ,1(R0)   ;THIS BIG
022342  112760  000002  000002                          MOVB    #RSSRD,2(R0)    ;OP CODE IS READ
022350  016560  000064  000012                          MOV     REC(R5),10.(R0) ;THIS RECORD
022356  116560  000060  000004                          MOVB    DR(R5),4.(R0)   ;THIS DRIVE
022364  112760  000000  000003                          MOVB    #0,3.(R0)       ;VERIFY
022372  012760  001000  000010                          MOV     #512.,8.(R0)    ;TOTAL BYTES TO READ
022400  112760  000020  000005                          MOVB    #020,5.(R0)     ;MAINTENANCE MODE
022406  005060  000006                                  CLR     6.(R0)          ;NO SEQUENCE #
022412  012701  000012                                  MOV     #RSMSIZ,R1      ;GET SIZE OF PACKET
022416  005721                                          TST     (R1)+           ;+2 FOR CHECKSUM
022420  012765  000016  000070                          MOV     #RSSNSZ,SNDCNT(R5)  ;SIZE TO SEND
022426  004737  013520                                  CALL    CHKSUM          ;FORM CHECKSUM R1=COUNT
022432  010110                                          MOV     R1,(R0)         ;INSERT CHECKSUM

022434  012701  001000                                  MOV     #512.,R1                ;SET EXPECTATIONS:
                                                                                ;CALC # OF DATA PACKETS TO EXPECT:
022440  012703  000034                                  MOV     #XSFLG,R3       ;GET OFFSET
022444  060503                                          ADD     R5,R3           ;ABS. ADDR. OF XSFLG
022446  005002                                          CLR     R2              ;PRESET AS NONE
022450  005202                                  64$:    INC     R2              ;# PACKETS EXPECTED
022452  012723  000001                                  MOV     #RSDATA,(R3)+   ;LOAD XSFLG
022456  012723  000204                                  MOV     #132.,(R3)+     ;AND EXPECTED COUNT
022462  162701  000200                                  SUB     #128.,R1        ;NEG RESULT LAST TIME
022466  101401                                          BLOS    66$             ;LAST TIME
022470  000767                                          BR      64$             ;MORE TO DO
022472  005202                                  66$:    INC     R2              ;ADD ONE FOR END PACK
```

```
022474  010265  000032                          MOV     R2,XSPKNM(R5)   ;SAVE # PACKETS TO EXPECT
022500  012723  000002                          MOV     #RSEND,(R3)+    ;EXPECT AN END ALSO...
022504  012713  000016                          MOV     #RSNDSZ,(R3)    ;THIS BIG-14. BYTES
022510  004737  006556                          CALL    RSVP            ;SEND
                                                                        ;AND RETURN TO SCHEDULER
022514  032715  002010                  67$:    BIT     #BIT10!BIT3,@R5 ;RETRY?
022520  001500                                  BEQ     65$             ;NO.
022522                                          TURTRY  REC(R5),#512.,DR(R5)    ;YES


022522  012700  026174                  72$:    MOV     #TRBUF,R0       ;FORM CMND PACK:
022526  112710  000002                          MOVB    #RSCMND,@R0     ;MESSAGE PACK TYPE
022532  112760  000012  000001                  MOVB    #RSMSIZ,1(R0)   ;THIS BIG
022540  112760  000002  000002                  MOVB    #RSSRD,2(R0)    ;OP CODE-READ
022546  016560  000064  000012                  MOV     REC(R5),10.(R0) ;THIS RECORD
022554  116560  000060  000004                  MOVB    DR(R5),4.(R0)   ;THIS DRIVE
022562  105060  000003                          CLRB    3(R0)           ;PRESET NORM THRESHOLD
022566  105715                                  TSTB    @R5             ;REDUCED?
022570  100002                                  BPL     73$             ;NO
022572  105260  000003                          INCB    3(R0)           ;YES-CHANGE THRESHOLD
022576  012760  001000  000010          73$:    MOV     #512.,8.(R0)    ;# BYTES DESIRED
022604  112760  000020  000005                  MOVB    #020,5.(R0)     ;MAINTENANCE MODE
022612  005060  000006                          CLR     6.(R0)          ;NO SEQUENCE #
022616  012701  000012                          MOV     #RSMSIZ,R1      ;SIZE OF PACKET
022622  005721                                  TST     (R1)+           ;PLUS FLAG+COUNT INTO R1
022624  012765  000016  000070                  MOV     #RSSNSZ,SNDCNT(R5)  ;SET UP SIZE TO SEND

022632  004737  013520                          CALL    CHKSUM          ;FORM CHECKSUM R1=COUNT
022636  010110                                  MOV     R1,(R0)         ;INSERT IN PACKET

022640  012701  001000                          MOV     #512.,R1                ;SET EXPECTATIONS:
                                                                        ;CALC # OF DATA PACKETS TO EXPECT
022644  012703  000034                          MOV     #XSFLG,R3       ;OFFSET OF FLAG
022650  060503                                  ADD     R5,R3           ;ABS. ADDR. OF XSFLG
022652  005002                                  CLR     R2              ;PRESET
022654  005202                          69$:    INC     R2              ;# PACKETS EXPECTED
022656  012723  000001                          MOV     #RSDATA,(R3)+   ;LOAD XSFLG
022662  012723  000204                          MOV     #132.,(R3)+     ;AND EXPECT COUNT
022666  162701  000200                          SUB     #128.,R1        ;NEG RESULT LAST TIME
022672  101401                                  BLOS    71$             ;LAST TIME!
022674  000767                                  BR      69$             ;MORE TO DO
022676  005202                          71$:    INC     R2              ;ADD ONE FOR END PACK
022700  010265  000032                          MOV     R2,XSPKNM(R5)   ;SAVE # PACKETS TO EXPECT
022704  012723  000002                          MOV     #RSEND,(R3)+    ;EXPECT AN END
022710  012713  000016                          MOV     #RSNDSZ,(R3)    ;THIS BIG-14. BYTES

022714  004737  006556                          CALL    RSVP            ;SEND
                                                                        ;AND RETURN TO SCHEDULER

3671 022724  005365  000066                  DEC     TMP(R5)         ;DO ALL RECORDS THIS TRACK?
3672 022730  001404                          BEQ     2$              ;YES-GET OTHER TRACK
3673 022732  005265  000064                  INC     REC(R5)         ;NO-NEXT RECORD
3674 022736  000137  022302                  JMP     1$              ;EXECUTE THE READ
3675 022742  005765  000062          2$:     TST     TRK(R5)         ;DONE 2 TRACKS?
3676 022746  001012                          BNE     TST5EX          ;YES-EXIT
3677 022750  005265  000062                  INC     TRK(R5)         ;NO-SET FLAG FOR NEXT PASS
3678 022754  013765  003332  000064          MOV     SECREC,REC(R5)  ;GET NEW STARTING BLOCK #
```

```
3679 022762  013765  003306  000066          MOV     TAPLEN,TMP(R5)  ;RESET # OF BLOCKS
3680 022770  000137  022302                  JMP     1$              ;AND EXECUTE
3681 022774  005237  003320          TST5EX: INC     DONE            ;DONE
3682 023000  000207                  RETURN                          ;RETURN
3683
3684 023002                          ENDTST
     023002                                                  L10020:
     023002  104401                                                 TRAP    CSETST
```

```
                                .SBTTL  TEST 6 / WRITE-VERIFY SELECTED NUMBER OF BLOCKS

                                        BGNTST
                                                                                T6::
                                        TSTID   #TST6
023004  012737  023050  003324                        MOV     #TST6,TSTTOP    ;SAVE ADDR OF TEST
023012  004737  006002                                CALL    SETUP           ;INIT UNITS TSTPC
023016  004737  005630                                CALL    SETDR           ;GET 1ST DRVS.
023022  004737  006050                                CALL    RUN             ;DO TEST
023026  004737  005526                                CALL    SWAPDR          ;GET NEXT DRVS.
023032  103004                                        BCC     64$             ;BR NO 2ND DRVS
023034  004737  006002                                CALL    SETUP           ;REINIT UNITS TSTPC
023040  004737  006050                                CALL    RUN             ;REPEAT TEST
023044                                          64$:                          ;DONE
                                        EXIT TST
023044  104432                                                        TRAP    C$EXIT
023046  000724                                                        .WORD   L10021-.

023050  005065  000064          TST6:   CLR     REC(R5)         ;START AT REC 0
023054  013765  003306  000066          MOV     TAPLEN,TMP(R5)  ;GET THE # OF BLOCKS PER TRACK
023062  005065  000062                  CLR     TRK(R5)         ;TRK(R5)=1ST OR 2ND PASS
023066  016565  000064  000072  1$:     MOV     REC(R5),PATTEN(R5)  ;USE RECORD NO. FOR DATA
023074  005737  002216                  TST     DRVCHK          ;ADD DR #?
023100  001403                          BEQ     10$             ;NO
023102  066565  000060  000072          ADD     DR(R5),PATTEN(R5)       ;ADD DRIVE ID
023110                          10$:    TUWRIT  PATTEN(R5),REC(R5),#512.,DR(R5),#1
023110  012700  026174                          72$:    MOV     #TRBUF,R0       ;MAKE COMMAND PACKET:
023114  112710  000002                          MOVB    #RSCMND,@R0     ;COMMAND FLAG
023120  112760  000012  000001                  MOVB    #RSMSIZ,1(R0)   ;THIS SIZE
023126  112760  000003  000002                  MOVB    #RSSWR,2(R0)    ;INSERT OP CODE-WRITE
023134  112760  000001  000003                  MOVB    #1,3.(R0)       ;VERIFY (1 OR 0)
023142  116560  000060  000004                  MOVB    DR(R5),4.(R0)   ;DRIVE #
023150  112760  000020  000005                  MOVB    #020,5.(R0)     ;MAINTENANCE MODE SWITCH
023156  005060  000006                          CLR     6.(R0)          ;NO SEQUENCE #
023162  012760  001000  000010                  MOV     #512.,8.(R0)    ;TOTAL COUNT TO WRITE
023170  016560  000064  000012                  MOV     REC(R5),10.(R0) ;AT  RECORD N
023176  012701  000012                          MOV     #RSMSIZ,R1      ;THE PACKET SIZE PLUS+2
023202  005721                                  TST     (R1)+           ;(FLAG AND COUNT) INTO R1
023204  012765  000016  000070                  MOV     #RSSNSZ,SNDCNT(R5)  ;LOAD THE  SIZE TO SEND
023212  004737  013520                          CALL    CHKSUM          ;R0 --> R1=COUNT
023216  010110                                  MOV     R1,(R0)         ;PUT CHKSUM IN PACKET
                                                                        ;SET UP EXPECTATIONS:
023220  012765  000020  000034                  MOV     #RSCONT,XSFLG(R5)   ;THE FLAG
023226  012765  000001  000036                  MOV     #1,XSCNT(R5)    ;THE COUNT
023234  012765  000001  000032                  MOV     #1,XSPKNM(R5)   ;THE # PACKETS EXPECTED
023242  012702  001000                          MOV     #512.,R2                ;GET # OF DATA BYTES
023246  004737  006556                          CALL    RSVP            ;SEND (AND RETURN TO SCHEDULER)
023252  032715  000010                          BIT     #BIT3,@R5       ;FLAG BYTE ERROR?
023256  001314                                  BNE     72$             ;YES
023260  042715  010000                          BIC     #BIT12,@R5      ;FLAG FOR LAST PACKET
023264  012700  026174          64$:    MOV     #TRBUF,R0       ;POINT TO TOP OF BUFFER AGAIN
023270  020227  000200                  CMP     R2,#128.        ;START DATA PACKET(S)
023274  101004                          BHI     65$             ;#512. > 128.!
023276  010201                          MOV     R2,R1           ;#512.<128.
023300  052715  010000                  BIS     #BIT12,@R5      ;SO LAST PACKET NOW
023304  000402                          BR      66$             ;USE REMAINING COUNT
```

```
023306  012701  000200                  65$:    MOV     #128.,R1        ;USE 128. BYTES
023312  110160  000001                  66$:    MOVB    R1,1(R0)        ;COPY COUNT TO BUFFER
023316  010103                                  MOV     R1,R3           ;R3=COUNTER TO LOAD BUFFER
023320  112710  000001                          MOVB    #RSDATA,@R0     ;FLAG FIRST
023324  005720                                  TST     (R0)+           ;SKIP COUNT
023326  116520  000072                  67$.    MOVB    PATTEN(R5),(R0)+        ;INSERT DATA
023332  005303                                  DEC     R3              ;MORE?
023334  101374                                  BHI     67$             ;YES
023336  012700  026174                          MOV     #TRBUF,R0       ;-->TOP AGAIN
023342  116001  000001                          MOVB    1(R0),R1        ;GET COUNT
023346  042701  177400                          BIC     #177400,R1      ;ZERO SIGN EXTEND
023352  010165  000070                          MOV     R1,SNDCNT(R5)   ;HOW MANY TO SEND PLUS
023356  062765  000004  000070                  ADD     #4,SNDCNT(R5)   ;FLAG,COUNT,CHKSUM
023364  062701  000002                          ADD     #2,R1           ;COMPENSATE FOR FLAG + COUNT
023370  004737  013520                          CALL    CHKSUM          ;FOR CHECKSUM CALC.
023374  110120                                  MOVB    R1,(R0)+        ;CHKSUM INTO PACKET
023376  000301                                  SWAB    R1              ;EVEN ON AN ODD
023400  110120                                  MOVB    R1,(R0)+        ;BYTE BOUNDARY
023402  032715  010000                          BIT     #BIT12,@R5      ;LAST DATA PACKET?
023406  001412                                  BEQ     68$             ;NO
023410  012765  000002  000034                  MOV     #RSEND,XSFLG(R5)  ;YES-EXPECT 'END'
023416  012765  000016  000036                  MOV     #RSNDSZ,XSCNT(R5)  ;OF THIS SIZE
023424  012765  000001  000032                  MOV     #1,XSPKNM(R5)   ;AND 1 PACKET
023432  000411                                  BR      69$             ;SEND
023434  012765  000020  000034          68$:    MOV     #RSCONT,XSFLG(R5)  ;(NOT LAST), EXPECT 'CONTINUE'
023442  012765  000001  000036                  MOV     #1,XSCNT(R5)    ;AND 1 BYTE
023450  012765  000001  000032                  MOV     #1,XSPKNM(R5)   ;AND 1 PACKET
023456  004737  006556                  69$:    CALL    RSVP            ;SEND PACKET
                                                                        ;AND RETURN TO SCHEDULER
023462  032715  000010                          BIT     #BIT3,@R5       ;FLAG BYTE RETRY?
023466  001210                                  BNE     72$             ;YES
023470  032715  002000                          BIT     #BIT10,@R5      ;RETRY DATA ERROR?
023474  001004                                  BNE     70$             ;YES
023476  162702  000200                          SUB     #128.,R2        ;NO, MORE DATA TO SEND?
023502  101270                                  BHI     64$             ;YES
023504  000502                                  BR      71$             ;NO
023506                          70$:    TURTRY  REC(R5),#512.,DR(R5)    ;RETRY HERE


023506  012700  026174                  76$:    MOV     #TRBUF,R0       ;FORM CMND PACK:
023512  112710  000002                          MOVB    #RSCMND,@R0     ;MESSAGE PACK TYPE
023516  112760  000012  000001                  MOVB    #RSMSIZ,1(R0)   ;THIS BIG
023524  112760  000002  000002                  MOVB    #RSSRD,2(R0)    ;OP CODE-READ
023532  016560  000064  000012                  MOV     REC(R5),10.(R0) ;THIS RECORD
023540  116560  000060  000004                  MOVB    DR(R5),4.(R0)   ;THIS DRIVE
023546  105060  000003                          CLRB    3(R0)           ;PRESET NORM THRESHOLD
023552  105715                                  TSTB    @R5             ;REDUCED?
023554  100002                                  BPL     77$             ;NO
023556  105260  000003                          INCB    3(R0)           ;YES-CHANGE THRESHOLD
023562  012760  001000  000010          77$:    MOV     #512.,8.(R0)    ;# BYTES DESIRED
023570  112760  000020  000005                  MOVB    #020,5.(R0)     ;MAINTENANCE MODE
023576  005060  000006                          CLR     6.(R0)          ;NO SEQUENCE #
023602  012701  000012                          MOV     #RSMSIZ,R1      ;SIZE OF PACKET
023606  005721                                  TST     (R1)+           ;PLUS FLAG+COUNT INTO R1
023610  012765  000016  000070                  MOV     #RSSNSZ,SNDCNT(R5)  ;SET UP SIZE TO SEND

023616  004737  013520                          CALL    CHKSUM          ;FORM CHECKSUM R1=COUNT
```

```
023622  010110                                          MOV     R1,(R0)         ;INSERT IN PACKET

023624  012701  001000                                  MOV     #512.,R1                ;SET EXPECTATIONS:
                                                                                ;CALC # OF DATA PACKETS TO EXPECT
023630  012703  000034                                  MOV     #XSFLG,R3       ;OFFSET OF FLAG
023634  060503                                          ADD     R5,R3           ;ABS. ADDR. OF XSFLG
023636  005002                                          CLR     R2              ;PRESET
023640  005202                                  73$:    INC     R2              ;# PACKETS EXPECTED
023642  012723  000001                                  MOV     #RSDATA,(R3)+   ;LOAD XSFLG
023646  012723  000204                                  MOV     #132.,(R3)+     ;AND EXPECT COUNT
023652  162701  000200                                  SUB     #128.,R1        ;NEG RESULT LAST TIME
023656  101401                                          BLOS    75$             ;LAST TIME!
023660  000767                                          BR      73$             ;MORE TO DO
023662  005202                                  75$:    INC     R2              ;ADD ONE FOR END PACK
023664  010265  000032                                  MOV     R2,XSPKNM(R5)   ;SAVE # PACKETS TO EXPECT
023670  012723  000002                                  MOV     #RSEND,(R3)+    ;EXPECT AN END
023674  012713  000016                                  MOV     #RSNDSZ,(R3)    ;THIS BIG-14. BYTES

023700  004737  006556                                  CALL    RSVP            ;SEND
                                                                                ;AND RETURN TO SCHEDULER

3702 023714  005365  000066                    DEC     TMP(R5)         ;DO ALL RECORDS FOR THIS TRACK?
3703 023720  001404                            BEQ     2$              ;YES-GET OTHER TRACK
3704 023722  005265  000064                    INC     REC(R5)         ;NO-NEXT RECORD
3705 023726  000137  023066                    JMP     1$              ;EXECUTE THE WRITE
3706 023732  005765  000062            2$:     TST     TRK(R5)         ;DONE 2 TRACKS?
3707 023736  001012                            BNE     TST6EX          ;YES-EXIT
3708 023740  005265  000062                    INC     TRK(R5)         ;NO-SET FLAG FOR NEXT PASS
3709 023744  013765  003332  000064            MOV     SECREC,REC(R5)  ;GET NEW STARTING BLOCK #
3710 023752  013765  003306  000066            MOV     TAPLEN,TMP(R5)  ;RESET # OF BLOCKS
3711 023760  000137  023066                    JMP     1$              ;AND EXECUTE
3712 023764  005237  003320            TST6EX: INC     DONE            ;DONE
3713 023770  000207                            RETURN                  ;RETURN
3714
3715 023772                                    ENDTST
     023772                                                     L10021:
     023772  104401                                                    TRAP    CSETST
```

```
                                .SBTTL  TEST 7 / READ-REDUCED THRESHOLD SELECTED NUMBER OF BLOCKS

                                        BGNTST
                                                                                T7::
                                        TSTID   #TST7
023774  012737  024040  003324                        MOV     #TST7,TSTTOP    ;SAVE ADDR OF TEST
024002  004737  006002                                CALL    SETUP           ;INIT UNITS TSTPC
024006  004737  005630                                CALL    SETDR           ;GET 1ST DRVS.
024012  004737  006050                                CALL    RUN             ;DO TEST
024016  004737  005526                                CALL    SWAPDR          ;GET NEXT DRVS.
024022  103004                                        BCC     64$             ;BR NO 2ND DRVS
024024  004737  006002                                CALL    SETUP           ;REINIT UNITS TSTPC
024030  004737  006050                                CALL    RUN             ;REPEAT TEST
024034                                          64$:                          ;DONE
                                        EXIT TST
024034  104432                                                        TRAP    C$EXIT
024036  000520                                                        .WORD   L10022-.

024040  005065  000064          TST7:   CLR     REC(R5)                 ;START AT REC 0
024044  013765  003306  000066          MOV     TAPLEN,TMP(R5)          ;GET THE # OF BLOCKS PER TRACK
024052  005065  000062                  CLR     TRK(R5)                 ;TRK(R5)=1ST OR 2ND PASS
024056  016565  000064  000072  1$:     MOV     REC(R5),PATTEN(R5)      ;USE RECORD NO. FOR DATA
024064  005737  002216                  TST     DRVCHK                  ;ADD DR #?
024070  001403                          BEQ     10$                     ;NO
024072  066565  000060  000072          ADD     DR(R5),PATTEN(R5)       ;ADD DRIVE ID
024100                          10$:    TUREAD  REC(R5),#512.,DR(R5),#1


024100  012700  026174                  68$:    MOV     #TRBUF,R0       ;FORM CMND PACK:
024104  112710  000002                          MOVB    #RSCMND,@R0     ;MESSAGE PACK TYPE
024110  112760  000012  000001                  MOVB    #RSMSIZ,1(R0)   ;THIS BIG
024116  112760  000002  000002                  MOVB    #RSSRD,2(R0)    ;OP CODE IS READ
024124  016560  000064  000012                  MOV     REC(R5),10.(R0) ;THIS RECORD
024132  116560  000060  000004                  MOVB    DR(R5),4.(R0)   ;THIS DRIVE
024140  112760  000001  000003                  MOVB    #1,3.(R0)       ;VERIFY
024146  012760  001000  000010                  MOV     #512.,8.(R0)    ;TOTAL BYTES TO READ
024154  112760  000020  000005                  MOVB    #020,5.(R0)     ;MAINTENANCE MODE
024162  005060  000006                          CLR     6.(R0)          ;NO SEQUENCE #
024166  012701  000012                          MOV     #RSMSIZ,R1      ;GET SIZE OF PACKET
024172  005721                                  TST     (R1)+           ;+2 FOR CHECKSUM
024174  012765  000016  000070                  MOV     #RSSNSZ,SNDCNT(R5)  ;SIZE TO SEND
024202  004737  013520                          CALL    CHKSUM          ;FORM CHECKSUM R1=COUNT
024206  010110                                  MOV     R1,(R0)         ;INSERT CHECKSUM

024210  012701  001000                          MOV     #512.,R1                ;SET EXPECTATIONS:
                                                                        ;CALC # OF DATA PACKETS TO EXPECT:
024214  012703  000034                          MOV     #XSFLG,R3       ;GET OFFSET
024220  060503                                  ADD     R5,R3           ;ABS. ADDR. OF XSFLG
024222  005002                                  CLR     R2              ;PRESET AS NONE
024224  005202                          64$:    INC     R2              ;# PACKETS EXPECTED
024226  012723  000001                          MOV     #RSDATA,(R3)+   ;LOAD XSFLG
024232  012723  000204                          MOV     #132.,(R3)+     ;AND EXPECTED COUNT
024236  162701  000200                          SUB     #128.,R1        ;NEG RESULT LAST TIME
024242  101401                                  BLOS    66$             ;LAST TIME
024244  000767                                  BR      64$             ;MORE TO DO
024246  005202                          66$:    INC     R2              ;ADD ONE FOR END PACK
```

```
024250  010265  000032                          MOV     R2,XSPKNM(R5)   ;SAVE # PACKETS TO EXPECT
024254  012723  000002                          MOV     #RSEND,(R3)+    ;EXPECT AN END ALSO...
024260  012713  000016                          MOV     #RSNDSZ,(R3)    ;THIS BIG-14. BYTES
024264  004737  006556                          CALL    RSVP            ;SEND
                                                                        ;AND RETURN TO SCHEDULER
024270  032715  002010                  67$:    BIT     #BIT10!BIT3,@R5 ;RETRY?
024274  001500                                  BEQ     65$             ;NO.
024276                                          TURTRY  REC(R5),#512.,DR(R5)    ;YES


024276  012700  026174                  72$:    MOV     #TRBUF,R0       ;FORM CMND PACK:
024302  112710  000002                          MOVB    #RSCMND,@R0     ;MESSAGE PACK TYPE
024306  112760  000012  000001                  MOVB    #RSMSIZ,1(R0)   ;THIS BIG
024314  112760  000002  000002                  MOVB    #RSSRD,2(R0)    ;OP CODE-READ
024322  016560  000064  000012                  MOV     REC(R5),10.(R0) ;THIS RECORD
024330  116560  000060  000004                  MOVB    DR(R5),4.(R0)   ;THIS DRIVE
024336  105060  000003                          CLRB    3(R0)           ;PRESET NORM THRESHOLD
024342  105715                                  TSTB    @R5             ;REDUCED?
024344  100002                                  BPL     73$             ;NO
024346  105260  000003                          INCB    3(R0)           ;YES-CHANGE THRESHOLD
024352  012760  001000  000010          73$:    MOV     #512.,8.(R0)    ;# BYTES DESIRED
024360  112760  000020  000005                  MOVB    #020,5.(R0)     ;MAINTENANCE MODE
024366  005060  000006                          CLR     6.(R0)          ;NO SEQUENCE #
024372  012701  000012                          MOV     #RSMSIZ,R1      ;SIZE OF PACKET
024376  005721                                  TST     (R1)+           ;PLUS FLAG+COUNT INTO R1
024400  012765  000016  000070                  MOV     #RSSNSZ,SNDCNT(R5)  ;SET UP SIZE TO SEND

024406  004737  013520                          CALL    CHKSUM          ;FORM CHECKSUM R1=COUNT
024412  010110                                  MOV     R1,(R0)         ;INSERT IN PACKET

024414  012701  001000                          MOV     #512.,R1                ;SET EXPECTATIONS:
                                                                        ;CALC # OF DATA PACKETS TO EXPECT
024420  012703  000034                          MOV     #XSFLG,R3       ;OFFSET OF FLAG
024424  060503                                  ADD     R5,R3           ;ABS. ADDR. OF XSFLG
024426  005002                                  CLR     R2              ;PRESET
024430  005202                          69$:    INC     R2              ;# PACKETS EXPECTED
024432  012723  000001                          MOV     #RSDATA,(R3)+   ;LOAD XSFLG
024436  012723  000204                          MOV     #132.,(R3)+     ;AND EXPECT COUNT
024442  162701  000200                          SUB     #128.,R1        ;NEG RESULT LAST TIME
024446  101401                                  BLOS    71$             ;LAST TIME!
024450  000767                                  BR      69$             ;MORE TO DO
024452  005202                          71$:    INC     R2              ;ADD ONE FOR END PACK
024454  010265  000032                          MOV     R2,XSPKNM(R5)   ;SAVE # PACKETS TO EXPECT
024460  012723  000002                          MOV     #RSEND,(R3)+    ;EXPECT AN END
024464  012713  000016                          MOV     #RSNDSZ,(R3)    ;THIS BIG-14. BYTES

024470  004737  006556                          CALL    RSVP            ;SEND
                                                                        ;AND RETURN TO SCHEDULER

3733 024500  005365  000066                      DEC     TMP(R5)         ;DO ALL RECORDS THIS TRACK?
3734 024504  001404                              BEQ     2$              ;YES-GET OTHER TRACK
3735 024506  005265  000064                      INC     REC(R5)         ;NO-NEXT RECORD
3736 024512  000137  024056                      JMP     1$              ;EXECUTE THE READ
3737 024516  005765  000062              2$:     TST     TRK(R5)         ;DONE 2 TRACKS?
3738 024522  001012                              BNE     TST7EX          ;YES-EXIT
3739 024524  005265  000062                      INC     TRK(R5)         ;NO-SET FLAG FOR NEXT PASS
3740 024530  013765  003332  000064              MOV     SECREC,REC(R5)  ;GET NEW STARTING BLOCK #
```

```
 3741 024536  013765  003306  000066          MOV     TAPLEN,TMP(R5)  ;RESET # OF BLOCKS
 3742 024544  000137  024056                  JMP     1$              ;AND EXECUTE
 3743 024550  005237  003320          TST7EX: INC     DONE            ;DONE
 3744 024554  000207                          RETURN                  ;RETURN
 3745
 3746 024556                                  ENDTST
      024556                                                      L10022:
      024556  104401                                                    TRAP    C$ETST
```

```
                                .SBTTL  TEST 8 / TESTS MODIFIED RADIAL SERIAL PROTOCOL

024560                          BGNTST
024560                                                                  T8::

024560  012737  024602  003324  MOV     #TST8,TSTTOP    ;SAVE ADDR OF TEST
024566  004737  006002          CALL    SETUP           ;INIT UNITS TSTPC
024572  004737  006050          CALL    RUN             ;DO TEST
                                ;DONE

024576                          EXIT TST
024576  104432                                                  TRAP    C$EXIT
024600  000662                                                  .WORD   L10023-.

024602  012737  000001  003340  TST8:   MOV     #1,TEST8        ;INDICATES 1ST PART OF TST 8
024610  012700  026174          64$:    MOV     #TRBUF,R0       ;FORM COMMAND PACKET
024614  112710  000002          MOVB    #RSCMND,@R0     ;COMMAND FLAG
024620  112760  000012  000001  MOVB    #RSMSIZ,1(R0)   ;SIZE OF MESSAGE
024626  112760  000012  000002  MOVB    #RSSGET,2(R0)   ;GET CHARACTERISTICS
024634  105060  000003          CLRB    3(R0)           ;NO MODIFIER.
024640  005060  000004          CLR     4(R0)           ;NO DRIVE OR SWITCHES
024644  005060  000006          CLR     6(R0)           ;NO SEQUENCE NUMBER
024650  005060  000010          CLR     8.(R0)          ;NO BYTES
024654  005060  000012          CLR     10.(R0)         ;NO RECORD #
024660  012701  000012          MOV     #RSMSIZ,R1      ;GET SIZE
024664  005721                  TST     (R1)+           ;+2 FOR CHECKSUM
024666  012765  000016  000070  MOV     #RSSNSZ,SNDCNT(R5)  ;SIZE TO SEND
024674  004737  013520          CALL    CHKSUM          ;FORM CHECKSUM
024700  010110                  MOV     R1,(R0)         ;INSERT INTO PACKET
024702  012765  000001  000034  MOV     #RSDATA,XSFLG(R5)   ;EXPECT DATA PACKET
024710  012765  000034  000036  MOV     #RSGCDP,XSCNT(R5)   ;THIS BIG
024716  012765  000001  000032  MOV     #1,XSPKNM(R5)   ;AND 1 PACKET
                                ;SEND
024724  004737  006556          CALL    RSVP            ;RETURN TO SCHEDULER

024730  004737  013654          CALL    DOBRK           ;CLR POTENTIAL INTERFACE ERROR

024734  032715  000010          BIT     #BIT3,@R5       ;RETRY?(BAD FLAG)
024740  001323                  BNE     64$             ;YES

024742  012737  000002  003340  MOV     #2,TEST8        ;INDICATE 2ND PART OF TST 8

024750  012700  026174  65$:    MOV     #TRBUF,R0       ;-->(POINT TO) XMIT BUFFER
024754  112710  000002          MOVB    #RSCMND,@R0     ;FORM COMMAND MESSAGE PACK
024760  112760  000012  000001  MOVB    #RSMSIZ,1(R0)   ;THIS BIG
024766  112760  000001  000002  MOVB    #RSSNIT,2(R0)   ;OP CODE IS INITIALIZE
024774  013760  000064  000012  MOV     REC,10.(R0)     ;TO THIS RECORD
025002  105060  000003          CLRB    3.(R0)          ;NO MODIFIER
025006  105060  000004          CLRB    4.(R0)          ;NO DRIVE
025012  112760  000010  000005  MOVB    #BIT03,5.(R0)   ;SET MRSP SWITCH
025020  005060  000006          CLR     6.(R0)          ;NO SEQUENCE #
025024  005060  000010          CLR     8.(R0)          ;NO BYTE COUNT
025030  012701  000012          MOV     #RSMSIZ,R1      ;GET COUNT
025034  005721                  TST     (R1)+           ;PLUS FLAG + BCNT
                                ;FOR CHECKSUM CALC
025036  004737  013520          CALL    CHKSUM          ;R0-->TOP   R1=# OF BYTES
025042  010110                  MOV     R1,(R0)         ;INSERT INTO PACKET
```

```
3803                                                    ;SET UP EXPECTATIONS:
3804 025044  012765  000016  000070        MOV     #RSSNSZ,SNDCNT(R5)  ;HOW MANY TO SEND
3805 025052  112765  000002  000034        MOVB    #RSCMND,XSFLG(R5)   ;EXPECT END PACK
3806 025060  012765  000016  000036        MOV     #RSNDSZ,XSCNT(R5)   ;COUNT WITH THIS
3807 025066  012765  000001  000032        MOV     #1.,XSPKNM(R5)  ;EXPECT ONLY 1 PACKET
3808
3809 025074  004737  006556                CALL    RSVP            ;SEND
3810                                                    ;AND RETURN TO SCHEDULER
3811
3812 025100  032715  000010                BIT     #BIT3,@R5       ;RETRY (FLAG BYTE ERROR)?
3813 025104  001321                        BNE     65$             ;YES
3814
3815 025106  012700  026174        66$:    MOV     #TRBUF,R0       ;-->(POINT TO) XMIT BUFFER
3816 025112  112710  000002                MOVB    #RSCMND,@R0     ;FORM COMMAND MESSAGE PACK
3817 025116  112760  000012  000001        MOVB    #RSMSIZ,1(R0)   ;THIS BIG
3818 025124  112760  000000  000002        MOVB    #RSSNOP,2(R0)   ;OP CODE IS NO-OPERATION
3819 025132  013760  000064  000012        MOV     REC,10.(R0)     ;TO THIS RECORD
3820 025140  105060  000003                CLRB    3.(R0)          ;NO MODIFIER
3821 025144  105060  000004                CLRB    4.(R0)          ;NO DRIVE
3822 025150  112760  000010  000005        MOVB    #BIT03,5.(R0)   ;SET MRSP SWITCH
3823 025156  005060  000006                CLR     6.(R0)          ;NO SEQUENCE #
3824 025162  005060  000010                CLR     8.(R0)          ;NO BYTE COUNT
3825 025166  012701  000012                MOV     #RSMSIZ,R1      ;GET COUNT
3826 025172  005721                        TST     (R1)+           ;PLUS FLAG + BCNT
3827                                                    ;FOR CHECKSUM CALC
3828 025174  004737  013520                CALL    CHKSUM          ;R0-->TOP   R1=# OF BYTES
3829 025200  010110                        MOV     R1,(R0)         ;INSERT INTO PACKET
3830                                                    ;SET UP EXPECTATIONS:
3831 025202  012765  000016  000070        MOV     #RSSNSZ,SNDCNT(R5)  ;HOW MANY TO SEND
3832 025210  112765  000002  000034        MOVB    #RSCMND,XSFLG(R5)   ;EXPECT END PACK
3833 025216  012765  000016  000036        MOV     #RSNDSZ,XSCNT(R5)   ;COUNT WITH THIS
3834 025224  012765  000001  000032        MOV     #1.,XSPKNM(R5)  ;EXPECT ONLY 1 PACKET
3835
3836 025232  004737  006556                CALL    RSVP            ;SEND
3837                                                    ;AND RETURN TO SCHEDULER
3838
3839 025236  032715  000010                BIT     #BIT3,@R5       ;RETRY (FLAG BYTE ERRGR)?
3840 025242  001321                        BNE     66$             ;YES
3841
3842 025244  005237  003320                INC     DONE
3843 025250  005037  003340                CLR     TEST8
3844
3845 025254  005737  002222                TST     PPSOT8          ;PROTOCOL SUMMARY @ END OF PASS
3846 025260  001477                        BEQ     ENDT8           ;NO
3847 025262  005037  025640                CLR     UNITNO          ;SET UNIT # TO ZERO
3848 025266                                PRINTF  #MSAGE1         ;PRINT HEADER
      025266  012746  025464                                                MOV     #MSAGE1,-(SP)
      025272  012746  000001                                                MOV     #1,-(SP)
      025276  010600                                                        MOV     SP,R0
      025300  104417                                                        TRAP    C$PNTF
      025302  062706  000004                                                ADD     #4,SP
3849 025306  012737  003346  003310        MOV     #BLKTBL,DEVPTR  ;SET ALL UNITS
3850 025314  017705  155770        1$:     MOV     @DEVPTR,R5      ;GET POINTER
3851 025320  005765  000000                TST     STATUS(R5)      ;IS UNIT ABORTED
3852 025324  100431                        BMI     3$              ;YES
3853 025326  005765  000210                TST     MRSP(R5)        ;IS UNIT MODIFIED
3854 025332  001413                        BEQ     2$              ;NO
```

```
                                        PRINTF  #MSAGE2,UNITNO  ;MESSAGE FOR MODIFIED UNIT
025334  013746  025640                                                    MOV     UNITNO,-(SP)
025340  012746  025525                                                    MOV     #MSAGE2,-(SP)
025344  012746  000002                                                    MOV     #2,-(SP)
025350  010600                                                            MOV     SP,R0
025352  104417                                                            TRAP    C$PNTF
025354  062706  000006                                                    ADD     #6,SP
                                        BR      4$              ;SEE IF LAST UNIT
                                2$:     PRINTF  #MSAGE3,UNITNO  ;MESSAGE FOR NON-MODIFIED UNIT
025362  013746  025640                                                    MOV     UNITNO,-(SP)
025366  012746  025561                                                    MOV     #MSAGE3,-(SP)
025372  012746  000002                                                    MOV     #2,-(SP)
025376  010600                                                            MOV     SP,R0
025400  104417                                                            TRAP    C$PNTF
025402  062706  000006                                                    ADD     #6,SP
                                        BR      4$              ;SEE IF LAST UNIT
                                3$:     PRINTF  #MSAGE4,UNITNO  ;MESSAGE FOR ABORTED UNIT
025410  013746  025640                                                    MOV     UNITNO,-(SP)
025414  012746  025610                                                    MOV     #MSAGE4,-(SP)
025420  012746  000002                                                    MOV     #2,-(SP)
025424  010600                                                            MOV     SP,R0
025426  104417                                                            TRAP    C$PNTF
025430  062706  000006                                                    ADD     #6,SP
025434  023727  003310  003364  4$:     CMP     DEVPTR,#LSTDEV  ;IS THIS THE LAST DEVICE
025442  103006                        BHIS    ENDT8           ;YES
025444  062737  000002  003310        ADD     #2,DEVPTR       ;GET NEXT UNIT
025452  005237  025640                INC     UNITNO          ;INC UNIT #
025456  000716                        BR      1$

025460  000207                ENDT8:  RETURN

025462                                  ENDTST
025462                                                  L10023:
025462  104401                                                  TRAP    C$ETST

025464      045     116     045  MSAGE1: .ASCIZ  /%N%S8%AUNIT NO%S9%S6%APROTOCOL%N/
025525      045     116     045  MSAGE2: .ASCIZ  !%N%S9%S2%01%S9%S9%ARSP/MRSP!
025561      045     116     045  MSAGE3: .ASCIZ  /%N%S9%S2%01%S9%S9%ARSP/
025610      045     116     045  MSAGE4: .ASCIZ  /%N%S9%S2%01%S9%S9%A---/
                                        .EVEN
025640  000000                UNITNO: .WORD
```

```
                                .SBTTL  PATCH AREA
000144                          .REPT   100.
                                .WORD
                                .ENDR
```

```
                                .SBTTL  I/O BUFFER AREAS:

                                ;WHO-GETS-WHAT-SPACE TABLE

026152  027232                  BUFTBL: .WORD   BUF0
026154  030270                          .WORD   BUF1
026156  031326                          .WORD   BUF2
026160  032364                          .WORD   BUF3
026162  033422                          .WORD   BUF4
026164  034460                          .WORD   BUF5
026166  035516                          .WORD   BUF6
026170  036554                          .WORD   BUF7


                                ;-----------------------------------------------------------------
                                ;ONLY 1 TRANSMIT BUFFER NECESSARY:

026172      023                         .BYTE RSXOFF
026173      023                         .BYTE RSXOFF    ;SEND XOFF BEFORE EVERY PACKET

026174                                  TRBUF:  .BLKB RCBFSZ
                                ;-----------------------------------------------------------------


027232                          BUF0:   .BLKB   RCBFSZ
030270                          BUF1:   .BLKB   RCBFSZ
031326                          BUF2:   .BLKB   RCBFSZ
032364                          BUF3:   .BLKB   RCBFSZ
033422                          BUF4:   .BLKB   RCBFSZ
034460                          BUF5:   .BLKB   RCBFSZ
035516                          BUF6:   .BLKB   RCBFSZ
036554                          BUF7:   .BLKB   RCBFSZ

                                ;-----------------------------------------------------------------

037612                                  ENDMOD
```

```
                                .TITLE PARAMETER CODING
                                .SBTTL  HARDWARE PARAMETER CODING SECTION
037612                                  BGNMOD
                                ;++
                                ; THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
                                ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                ; WITH THE OPERATOR.
                                ;--
037612                                  BGNHRD
037612  000021                                                  .WORD L10024-L$HARD/2
037614                                                  L$HARD::
037614                                  GPRMA   MSG1,0,0,160000,177777,YES
037614  000031                                                  .WORD   T$CODE
037616  037656                                                  .WORD   MSG1
037620  160000                                                  .WORD   T$LOLIM
037622  177777                                                  .WORD   T$HILIM
037624                                  GPRMA   MSG1B,2,0,0,776,YES
037624  001031                                                  .WORD   T$CODE
037626  037667                                                  .WORD   MSG1B
037630  000000                                                  .WORD   T$LOLIM
037632  000776                                                  .WORD   T$HILIM
037634                                  GPRML   MSG1C,6,1,YES
037634  003130                                                  .WORD   T$CODE
037636  037704                                                  .WORD   MSG1C
037640  000001                                                  .WORD   1
037642                                  GPRML   MSG2,4,1,YES
037642  002130                                                  .WORD   T$CODE
037644  037722                                                  .WORD   MSG2
037646  000001                                                  .WORD   1
037650                                  GPRML   MSG3,4,2,YES
037650  002130                                                  .WORD   T$CODE
037652  037737                                                  .WORD   MSG3
037654  000002                                                  .WORD   2

037656                                  ENDHRD
                                                        .EVEN
037656                                                  L10024:
037656  124  125  065  MSG1:   .ASCIZ  /TU58 CSR/
037667  126  105  103  MSG1B:  .ASCIZ  /VECTOR ADDR./
037704  120  104  124  MSG1C:  .ASCIZ  /PDT INTERFACE/
037722  124  105  123  MSG2:   .ASCIZ  /TEST DRIVE 0/
037737  124  105  123  MSG3:   .ASCIZ  /TEST DRIVE 1/
                                .EVEN
```

```
                                .SBTTL  SOFTWARE PARAMETER CODING SECTION
                                ;++
                                ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
                                ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                ; WITH THE OPERATOR.
                                ;--

                                        BGNSFT
037754  000031                                                            .WORD L10025-L$SOFT/2
037756                                                                  L$SOFT::
                                        GPRMD   MSG4,0,D,1777,8.,512.,YES
037756  000052                                                          .WORD   T$CODE
037760  040040                                                          .WORD   MSG4
037762  001777                                                          .WORD   1777
037764  000010                                                          .WORD   T$LOLIM
037766  001000                                                          .WORD   T$HILIM
                                        GPRML   MSG4B,10,1,YES
037770  004130                                                          .WORD   T$CODE
037772  040105                                                          .WORD   MSG4B
037774  000001                                                          .WORD   1
                                        GPRML   MSG5,2,1,YES
037776  001130                                                          .WORD   T$CODE
040000  040147                                                          .WORD   MSG5
040002  000001                                                          .WORD   1
                                        GPRML   MSG6,6,1,YES
040004  003130                                                          .WORD   T$CODE
040006  040201                                                          .WORD   MSG6
040010  000001                                                          .WORD   1
                                        GPRML   MSG7,4,1,YES
040012  002130                                                          .WORD   T$CODE
040014  040226                                                          .WORD   MSG7
040016  000001                                                          .WORD   1
                                        GPRMD   MSG8,10.,D,377,1,254.,YES
040020  005052                                                          .WORD   T$CODE
040022  040254                                                          .WORD   MSG8
040024  000377                                                          .WORD   377
040026  000001                                                          .WORD   T$LOLIM
040030  000376                                                          .WORD   T$HILIM
                                        GPRML   MSG9,12.,1,YES
040032  006130                                                          .WORD   T$CODE
040034  040315                                                          .WORD   MSG9
040036  000001                                                          .WORD   1
                                SFTOUT: ENDSFT
                                                                        .EVEN
040040                                                                  L10025:
                        116     125     115     MSG4:   .ASCIZ  'NUMBER OF BLOCKS:TEST 4-7 (8 TO 512)'
                        101     104     104     MSG4B:  .ASCIZ  /ADD DR # TO DATA PATTERN:TEST 4-7/
                        123     124     101     MSG5:   .ASCIZ  /STATISTICS PRINTED AT EOP/
                        103     117     115     MSG6:   .ASCIZ  /COMPARE DATA ON READ/
                        120     122     111     MSG7:   .ASCIZ  /PRINT PACKET ON ERROR/
                        043     040     105     MSG8:   .ASCIZ  /# ERRORS = DVC FATAL IF 'EVL'SET/
                        120     122     111     MSG9:   .ASCIZ  /PRINT UNIT PROTOCOL SUMMARY (TEST 8)/
                                        .EVEN
```

```
 4063          000016                          .REPT   14.             ;LASTAD CORRECTION
 4064                                          .WORD
 4065                                          .ENDR
 4072 040416                                   LASTAD
                                                                                .EVEN
      040416  040436                                                           .WORD T$FREE
      040420  000006                                                           .WORD T$SIZE
      040422                          L$LAST::
 4073 040422                                   ENDMOD
 4074
 4075 040422                                   BGNSETUP        1
 4076 040422                                   BGNPTAB
      040422  000000                                                           .WORD   0
      040424  000004                                                           .WORD   L10030-./2-1
      040426                                                           L10026:
 4077 040426  176500                           176500
 4078 040430  000300                           300
 4079 040432  000003                           3
 4080 040434  000000                           0
 4081 040436                                   ENDPTAB
      040436                                                           L10030:
 4082 040436                                   ENDSETUP
 4083          000001                          .END
```

```
ABNDX = 000004 G    CHKANS  010442 G    C$MSG = 000023      ESTRY = 000001 G    G$PRMA= 000001
ABO     013012      CHKEND  011146 G    C$OPEN= 000034      ESWLOC= 177765 G    G$PRMD= 000002
ABOMSG  017120      CHKERR  011252      C$PNTB= 000014      ESWR  = 177757        G$PRML= 000000
ABONM   006530      CHKPKS  010552 G    C$PNTF= 000017      EVL   = 000004 G    G$RADA= 000140
ADR   = 000020 G    CHKPTR  010550      C$PNTS= 000016      EVLTHR  002220        G$RADB= 000000
ALLGON  003336 G    CHKREE  011324      C$PNTX= 000015      EXOFF   007425        G$RADD= 000040
ASSEMB= 000010      CHKRET  011670      C$QIO = 000377      EXON    007424        G$RADL= 000120
BDATA = 000134 G    CHKSUC  012164 G    C$RDBU= 000007      ESEND = 002100        G$RADO= 000020
BDBYTS  014572      CHKSUM  013520 G    C$REFG= 000047      ESLOAD= 000035        G$XFER= 000004
BDCHK = 000022 G    CHK8    010462      C$RESE= 000033      FLGLOC  016616 G    G$YES = 000010
BDCOM = 000014 G    CKCKSM  013614 G    C$REVI= 000003      FM      015574        HARDR = 000136 G
BIT0  = 000001 G    CLRALL  005706 G    C$RFLA= 000021      FM0     015556        HARDW = 000140 G
BIT00 = 000001 G    CLRBUF  005746 G    C$RPT = 000025      FTLNM   003316        HELP  = 000000
BIT01 = 000002 G    CLRPTR  006000      C$SEFG= 000046      F$AU  = 000015        HOE   = 100000 G
BIT02 = 000004 G    CMDSNT= 000100 G    C$SPRI= 000041      F$AUTO= 000020        HRD     012772
BIT03 = 000010 G    CMNDER= 000040 G    C$SVEC= 000037      F$BGN = 000040        HRDRD = 000016 G
BIT04 = 000020 G    CMPDAT  002214      C$TPRI= 000013      F$CLEA= 000007        HRDWR = 000020 G
BIT05 = 000040 G    CNINIT= 000032 G    DESC    014574      F$DU  = 000016        HRD1    011612
BIT06 = 000100 G    COMPAR  014414 G    DEVPTR  003310 G    F$END = 000041        IBE   = 010000 G
BIT07 = 000200 G    CSNRDY  003342 G    DEV0    003366      F$HARD= 000004        IDPTR   003322 G
BIT08 = 000400 G    CSRCVB  003344 G    DEV1    003600      F$HW  = 000013        IDU   = 000040 G
BIT09 = 001000 G    C$AU  = 000052      DEV2    004012      F$INIT= 000006        IER   = 020000 G
BIT1  = 000002 G    C$AUTO= 000061      DEV3    004224      F$JMP = 000050        INIT    016030
BIT10 = 002000 G    C$BRK = 000022      DEV4    004436      F$MOD = 000000        INITWD  014406
BIT11 = 004000 G    C$BSEG= 000004      DEV5    004650      F$MSG = 000011        INIT2   016056
BIT12 = 010000 G    C$BSUB= 000002      DEV6    005062      F$PROT= 000021        ISR   = 000100 G
BIT13 = 020000 G    C$CEFG= 000045      DEV7    005274      F$PWR = 000017        IXE   = 004000 G
BIT14 = 040000 G    C$CLCK= 000062      DFPTBL  002174 G    F$RPT = 000012        I$AU  = 000041
BIT15 = 100000 G    C$CLEA= 000012      DFTL1   012640      F$SEG = 000003        I$AUTO= 000041
BIT2  = 000004 G    C$CLOS= 000035      DIAGMC= 000000      F$SOFT= 000005        I$CLN = 000041
BIT3  = 000010 G    C$CLP1= 000006      DLV   = 000074 G    F$SRV = 000010        I$DU  = 000041
BIT4  = 000020 G    C$CVEC= 000036      DOBRK   013654 G    F$SUB = 000002        I$HRD = 000041
BIT5  = 000040 G    C$DCLN= 000044      DONE    003320 G    F$SW  = 000014        I$INIT= 000041
BIT6  = 000100 G    C$DODU= 000051      DR    = 000060 G    F$TEST= 000001        I$MOD = 000041
BIT7  = 000200 G    C$DRPT= 000024      DRVCHK  002216      GBTMP   010436        I$MSG = 000041
BIT8  = 000400 G    C$DU  = 000053      EF.CON= 000036 G    GBTMP2  010440        I$PROT= 000040
BIT9  = 001000 G    C$EDIT= 000003      EF.NEW= 000035 G    GETANS  007150 G    I$PTAB= 000041
BLKEND= 000202 G    C$ERDF= 000055      EF.PWR= 000034 G    GETHRD  016162        I$PWR = 000041
BLKER   003330 G    C$ERHR= 000056      EF.RES= 000037 G    GETPTR  007214        I$RPT = 000041
BLKSIZ= 000212 G    C$ERRO= 000060      EF.STA= 000040 G    GETR5   017066        I$SEG = 000041
BLKTBL  003346 G    C$ERSF= 000054      ENDGP8  010114      GTAGIN  007462        I$SETU= 000041
BOE   = 000400 G    C$ERSO= 000057      ENDRSP  007076      GTBYTE  010212 G    I$SFT = 000041
BRKPTR  014412      C$ESCA= 000010      ENDT8   025460      GTDOWN  010052        I$SRV = 000041
BRKTO   014410      C$ESEG= 000005      ERRDES  013034 G    GTOK    007720        I$SUB = 000041
BRKWD   014404      C$ESUB= 000003      ESABO = 177720 G    GTPKS1  007216 G    I$TST = 000041
BUFTBL  026152      C$ETST= 000001      ESCKS = 177757 G    GTPKS8  007426 G    J$JMP = 000167
BUF0    027232      C$EXIT= 000032      ESCKSM= 177757      GTPTR   010210        LENGTH  002206
BUF1    030270      C$GETB= 000026      ESCMD = 177720 G    GTUM    007662        LGOFST= 000120 G
BUF2    031326      C$GETW= 000027      ESNCRT= 177767 G    G$CNTO= 000200        LNCNT   014774
BUF3    032364      C$GMAN= 000043      ESNOMO= 177737 G    G$DELM= 000372        LOE   = 040000 G
BUF4    033422      C$GPHR= 000042      ESNONX= 177770 G    G$DISP= 000003        LOG     012500 G
BUF5    034460      C$GPLO= 000030      ESOK  = 000000 G    G$EXCP= 000400        LOGO    013022
BUF6    035516      C$GPRI= 000040      ESPART= 177776 G    G$HILI= 000002        LOGOK   012566
BUF7    036554      C$INIT= 000011      ESRD  = 177757      G$LOLI= 000001        LOGOK2  012652
CARLF   015010      C$INLP= 000020      ESREC = 177711 G    G$NO  = 000000        LOGO    012554
CHECK   016140      C$MANI= 000050      ESSK  = 177740 G    G$OFFS= 000400        LOG1    012670
CHKANR  010546      C$MEM = 000031      ESSLF = 177777 G    G$OFSI= 000376        LOG2    012720
```

```
LOG3     012760          L10001  002204          MSQRSP  002776 G        PTR     017116          SFTOUT  040040
LOG3B    013002          L10002  002224          MSREC   002716 G        RCBCNT  003314          SFTRD = 000002 G
LOT   = 000010 G         L10003  013224          MSRNIT  002534 G        RCBFSZ= 001036 G        SFTWR = 000004 G
LSTDEV   003364 G        L10004  014312          MSSELF  002362 G        RCDB  = 000024 G        SKERR = 000024 G
L$ACP    002110 G        L10005  014346          MSSFRD  003052 G        RCFLG   003312 G        SLFER = 000044 G
L$APT    002036 G        L10006  015436          MSSFWR  003112 G        RCINIT= 000006 G        SND     006646
L$AU     017146 G        L10007  016526          MSSKER  002304 G        RCSR  = 000022 G        SNDBYT  007100 G
L$AUT    002070 G        L10010  016724          MSTOSN  003030 G        RCVBUF= 000102 G        SNDCNT= 000070 G
L$AUTO   016620 G        L10011  017020          MSUNIT  002654 G        RCVHND  014314          SNDHND  014300
L$CCP    002106 G        L10012  017064          MSWPRO  002512 G        RCVINT  014314 G        SNDINT  014300 G
L$CLEA   017002 G        L10013  017146          MSWRSP  002736 G        RDN0  = 000114 G        SOFTR = 000122 G
L$CO     002032 G        L10014  017350          MXRTRY  003326 G        RDN1  = 000116 G        SOFTW = 000124 G
L$DEPO   002011 G        L10015  017622          NCART = 000054 G        REC   = 000064 G        SRVTBL  010166
L$DESC   002122 G        L10016  021226          NODRVS  016560          RECDAT  017606          STAEOP  002210
L$DESP   002076 G        L10017  022216          NOMOR   006532          RECERR= 000042 G        STATHD  015444
L$DEVP   002060 G        L10020  023002          NOMOT = 000030 G        RECID   013306 G        STATUS= 000000 G
L$DISP   002152 G        L10021  023772          NOREE   011200          RECID2  013462          STHD2   015720
L$DLY    002116 G        L10022  024556          NOUNIT= 000036 G        RECOV   011710          STRT    016612 G
L$DTP    002040 G        L10023  025462          NOXOFF  006630          RETERR  012112          SUCCS = 000076 G
L$DTYP   002034 G        L10024  037656          NTSFT   012702          RETRY = 000002 G        SUCOTL= 000046 G
L$DU     017022 G        L10025  040040          NXTRET  006526          RLUN    015440          SVCGBL= 000000
L$DUT    002072 G        L10026  040426          NXTST   006100 G        RPTR    015442          SVCINS= 000001
L$DVTY   005506 G        L10030  040436          NXTST2  006350          RSCMND= 000002 G        SVCSUB= 000001
L$EF     002052 G        MABEE   012732          ONEFIL= 000001          RSCONT= 000020 G        SVCTAG= 000001
L$ENVI   002044 G        MODRSP  010116          OTL   = 000010 G        RSDASZ= 000204 G        SVCTST= 000001
L$ETP    002102 G        MRSDLY  010120          OVRFLO  013400          RSDATA= 000001 G        SWAPDR  005526 G
L$EXP1   002046 G        MRSP  = 000210 G        OVRN  = 000012 G        RSDNSZ= 000222 G        SWPTR   005626
L$EXP4   002064 G        MSAGE1  025464          O$APTS= 000000          RSEND = 000002 G        SYSTAT  003304 G
L$EXP5   002066 G        MSAGE2  025525          O$AU  = 000001          RSGCDP= 000034 G        S$LSYM= 010000
L$HARD   037614 G        MSAGE3  025561          O$BGNR= 000001          RSINIT= 000004 G        TAPLEN  003306 G
L$HIME   002120 G        MSAGE4  025610          O$BGNS= 000001          RSMSIZ= 000012 G        TEST8   003340 G
L$HPCP   002016 G        MSAUTO  016762          O$DU  = 000001          RSNDSZ= 000016 G        THRSHI  012016
L$HPTP   002022 G        MSBDA   002336 G        O$ERRT= 000000          RSNTAB  002224          THRSLO  011770
L$HW     002174 G        MSCMD   002702 G        O$GNSW= 000001          RSSEND= 000100 G        TMP   = 000066 G
L$ICP    002104 G        MSCOM   002402 G        O$POIN= 000001          RSSGET= 000012 G        TOMANY  016530
L$INIT   016030 G        MSG1    037656          O$SETU= 000001          RSSNIT= 000001 G        TORCVB= 000050 G
L$LADP   002026 G        MSG1B   037667          PARTL = 000034 G        RSSNOP= 000000 G        TOSNDB= 000056 G
L$LAST   040422 G        MSG1C   037704          PATTEN= 000072 G        RSSNSZ= 000016 G        TRBUF   026174
L$LOAD   002100 G        MSG2    037722          PDTFLG  016614 G        RSSRD = 000002 G        TRK   = 000062 G
L$LUN    002074 G        MSG3    037737          PERDEV  006362          RSSSEK= 000005 G        TRPHND  016730
L$MREV   002050 G        MSG4    040040          PKPTR = 000104 G        RSSSLF= 000007 G        TRPPTR  016726
L$NAME   002000 G        MSG4B   040105          PNT   = 001000 G        RSSWR = 000003 G        TSTPC = 000020 G
L$PRIO   002042 G        MSG5    040147          PPSOT8  002222          RSVP    006556 G        TSTTOP  003324
L$PROT   002142 G        MSG6    040201          PRBUF   002212          RSXOFF= 000023 G        TST1    017214
L$PRT    002112 G        MSG7    040226          PRDAT   014776          RSXON = 000020 G        TST2    017416
L$REPP   002062 G        MSG8    040254          PRFORM  015000          RTRYN   012050          TST3    017670
L$REV    002010 G        MSG9    040315          PRI   = 002000 G        RUN     006050 G        TST3PT  021214
L$RPT    015014 G        MSHCHK  002554 G        PRI00 = 000000 G        SAVCNT= 000206 G        TST4    021274
L$SOFT   037756 G        MSHDRD  003152 G        PRI01 = 000040 G        SECREC  003332 G        TST4EX  022210
L$SPC    002056 G        MSHDWR  003214 G        PRI02 = 000100 G        SERVST  010206          TST5    022264
L$SPCP   002020 G        MSNIT   002616 G        PRI03 = 000140 G        SETDR   005630 G        TST5EX  022774
L$SPTP   002024 G        MSNLOG  002320 G        PRI04 = 000200 G        SETLEN  016470          TST6    023050
L$STA    002030 G        MSNOMO  002444 G        PRI05 = 000240 G        SETPTR  005704          TST6EX  023764
L$SW     002206 G        MSNOTP  002462 G        PRI06 = 000300 G        SETSRV  010122          TST7    024040
L$TEST   002114 G        MSNRSP  002762 G        PRI07 = 000340 G        SETUP   006002 G        TST7EX  024550
L$TIML   002014 G        MSOVRN  003256 G        PRNPAK  014630 G        SFPTBL  002206 G        TST8    024602
L$UNIT   002012 G        MSPART  002632 G        PRNSIZ  003334 G        SFT     012750          TUVECT= 000204 G
```

PARAMETER CODING MACRO M1113 25-SEP-81 10:06 PAGE 126-3 SEQ 0122
SYMBOL TABLE

```
T$ARGC= 000002          T$PTAB= 010027          T$$DAT= 010030          T2      017352 G        WHCHDR  013504 G
T$CODE= 006130          T$PTHV= 000001          T$$DU = 010012          T3      017624 G        WRLOCK= 000026 G
T$ERRN= 000146          T$PTNU= 000001          T$$HAR= 010024          T4      021230 G        WRTNO = 000110 G
T$EXCP= 000000          T$SAVL= 177777          T$$HW = 010001          T4TRY = 000132 G        WRTN1 = 000112 G
T$FLAG= 000040          T$SEGL= 177777          T$$INI= 010007          T5      022220 G        XFNSND  006636
T$FREE= 040436          T$SIZE= 000006          T$$MSG= 010003          T6      023004 G        XMDB  = 000030 G
T$GMAN= 000000          T$SUBN= 000000          T$$PC = 000001          T7      023774 G        XMSR  = 000026 G
T$HILI= 000376          T$TAGL= 177777          T$$PRO= 010000          T8      024560 G        XSCNT = 000036 G
T$LAST= 000001          T$TAGN= 010C31          T$$PTA= 010027          UAM   = 000200 G        XSFLG = 000034 G
T$LOLI= 000001          T$TEMP= 000000          T$$RPT= 010006          UNIT    013226 G        XSPKNM= 000032 G
T$LSYM= 010000          T$TEST= 000010          T$$SOF= 010025          UNITNO  025640          XSPTR = 000106 G
T$LTNO= 000010          T$TSTM= 177777          T$$SRV= 010005          UNREC   012070          X$ALWA= 000000
T$NEST= 177777          T$TSTS= 000001          T$$SW = 010002          UNSUC   011466          X$FALS= 000040
T$NS0 = 000000          T$$AU = 010013          T$$TES= 010023          UNXPCT  007624          X$OFFS= 000400
T$NS1 = 000005          T$$AUT= 010010          T1      017150 G        WAIT    014350          X$TRUE= 000020
T$PCNT= 000000          T$$CLE= 010011          T1TRY = 000146 G

. ABS.  040436     000
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  33777 WORDS  ( 132 PAGES)
DYNAMIC MEMORY:  21558 WORDS  ( 82 PAGES)
ELAPSED TIME:  00:06:01
CZTUUC.BIN/EN:AMA:ABS,CZTUUC/CR/SP=LB1:[1,1]SVC/MLB,SY:[203,375]CZTUUC.MAC
```

SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES								
ABNDX	= 000004	G	#16-888	*72-2659	72-2672	72-2674					
ABO	013012		72-2669	72-2683	72-2696	#72-2712					
ABOMSG	017120		97-3413	#97-3443							
ABONM	006530		*49-1799	*49-1816	49-1821	#49-1829					
ADR	= 000020	G	#10-640								
ALLGON	003336	G	#15-857	*49-1825	*91-3278	95-3370					
ASSEMB	= 000010		5-451	5-451							
BDATA	= 000134	G	#17-940	89-3153	89-3163						
BDBYTS	014572		*86-3001	*86-3012	86-3015	86-3020	#86-3030				
BDCHK	= 000022	G	#11-668	66-2365	66-2422						
BDCOM	= 000014	G	#11-665	68-2472	72-2684						
BIT0	= 000001	G	#10-640	37-1615	60-2193	72-2675	78-2786	78-2787	78-2789	78-2800	82-2875
BIT00	= 000001	G	#10-640	10-640							
BIT01	= 000002	G	#10-640	10-640							
BIT02	= 000004	G	#10-640	10-640							
BIT03	= 000010	G	#10-640	10-640	116-3795	116-3822					
BIT04	= 000020	G	#10-640	10-640							
BIT05	= 000040	G	#10-640	10-640							
BIT06	= 000100	G	#10-640	10-640							
BIT07	= 000200	G	#10-640	10-640							
BIT08	= 000400	G	#10-640	10-640							
BIT09	= 001000	G	#10-640	10-640							
BIT1	= 000002	G	#10-640	49-1781	54-1972	60-2194	64-2294	68-2458	68-2529		
BIT10	= 002000	G	#10-640	49-1779	50-1901	68-2478	68-2498	68-2506	68-2527	106-3596	106-3596
			106-3597	108-3639	108-3639	110-3670	112-3701	112-3701	114-3732		
BIT11	= 004000	G	#10-640	89-3139	91-3215	91-3236	93-3337				
BIT12	= 010000	G	#10-640	106-3596	106-3596	106-3596	108-3639	108-3639	108-3639	112-3701	112-3701
			112-3701								
BIT13	= 020000	G	#10-640	49-1769	49-1777	66-2381	66-2393	68-2462			
BIT14	= 040000	G	#10-640	49-1805	49-1808	70-2595	82-2925	91-3242			
BIT15	= 100000	G	#10-640	37-1613	49-1788	56-2014	56-2036	58-2079	58-2098	58-2118	58-2142
			64-2302	68-2464	91-3214	91-3261	93-3337	97-3410			
BIT2	= 000004	G	#10-640	49-1776	49-1790	54-1972	60-2195	64-2294	68-2458	68-2460	68-2529
BIT3	= 000010	G	#10-640	49-1767	60-2196	66-2345	66-2381	66-2393	102-3547	104-3565	106-3596
			106-3596	106-3596	106-3597	108-3639	108-3639	108-3639	110-3670	112-3701	112-3701
			112-3701	114-3732	116-3783	116-3812	116-3839				
BIT4	= 000020	G	#10-640	50-1895	50-1899	60-2197	74-2731	82-2877			
BIT5	= 000040	G	#10-640	50-1917	50-1919	60-2198					
BIT6	= 000100	G	#10-640	60-2199	68-2469	68-2530	82-2902	82-2910	82-2917	82-2933	82-2934
			84-2946	84-2954	86-3021						
BIT7	= 000200	G	#10-640	60-2200	68-2498	68-2506	68-2512	68-2514	68-2516	68-2527	
BIT8	= 000400	G	#10-640	39-1648	49-1771	50-1905	50-1914	68-2491	68-2520	91-3243	
BIT9	= 001000	G	#10-640	37-1617	50-1889	68-2475	68-2502	70-2585	91-3245		
BLKEND	= 000202	G	#17-960	91-3253							
BLKER	003330	G	#15-854	*50-1886	*91-3286						
BLKSIZ	= 000212	G	#17-964	19-986	19-987	19-988	19-989	19-990	19-991	19-992	19-993
BLKTBL	003346	G	#19-974	37-1611	39-1645	41-1670	45-1713	49-1763	49-1800	58-2077	64-2300
			89-3133	91-3210	91-3230	93-3335	97-3434	116-3849			
BOE	= 000400	G	#10-640								
BRKPTR	014412		*82-2901	*82-2909	*82-2916	84-2967	#84-2981				
BRKTO	014410		*82-2874	*82-2883	*82-2889	*82-2899	*82-2907	*84-2971	#84-2980		
BRKWD	014404		82-2888	#84-2977							

```
SYMBOL  VALUE            REFERENCES
BUFTBL    026152          91-3217   #120-3888
BUF0      027232          120-3888  #120-3908
BUF1      030270          120-3889  #120-3909
BUF2      031326          120-3890  #120-3910
BUF3      032364          120-3891  #120-3911
BUF4      033422          120-3892  #120-3912
BUF5      034460          120-3893  #120-3913
BUF6      035516          120-3894  #120-3914
BUF7      036554          120-3895  #120-3915
CARLF     015010          88-3059    88-3061   #88-3070
CHECK     016140          91-3220   #91-3225
CHKANR    010546          64-2298    64-2310   #64-2314
CHKANS    010442  G       47-1738   #64-2292
CHKEND    011146  G       66-2370    66-2418   #68-2455
CHKERR    011252          68-2468   #68-2475
CHKPKS    010552  G       64-2296    64-2308   #66-2344
CHKPTR    010550          *64-2300   64-2301    64-2309   *64-2311   #64-2316
CHKREE    011324          68-2461   #68-2482
CHKRET    011670          68-2466    68-2471    68-2474    68-2477    68-2481    68-2499    68-2507    68-2515    68-2517
                         #68-2529
CHKSUC    012164  G       68-2463    68-2482   #70-2564
CHKSUM    013520  G      #78-2784    80-2827    102-3547   104-3565   106-3596   106-3596   106-3596   106-3597   106-3597
                          108-3639   108-3639   108-3639   110-3670   110-3670   112-3701   112-3701   112-3701   114-3732
                          114-3732   116-3773   116-3801   116-3828
CHK8      010462          64-2295   #64-2300
CKCKSM    013614  G       66-2363    66-2401    66-2413   #80-2824
CLRALL    005706  G      #41-1670    54-1975
CLRBUF    005746  G       41-1672   #43-1688    54-1978
CLRPTR    006000          *41-1670    41-1671    41-1673   *41-1675   #43-1698
CMDSNT  = 000100  G      #16-913     *50-1898    70-2571    70-2575    70-2588    74-2737   *82-2876
CMNDER  = 000040  G      #11-675     70-2609
CMPDAT    002214         #9-566      86-3003
CNINIT  = 000032  G      #11-672     82-2929
COMPAR    014414  G       66-2379    66-2403   #86-2998
CSNRDY    003342  G      #15-864     52-1943
CSRCVB    003344  G      #15-865     62-2230
C$AU    = 000052         #5-451      99-3478
C$AUTO  = 000061         #5-451      93-3346
C$BRK   = 000022         #5-451      47-1736    49-1824    52-1947    62-2252    82-2879    84-2969    84-2970    89-3132
                          89-3135    89-3137
C$BSEG  = 000004         #5-451
C$BSUB  = 000002         #5-451
C$CEFG  = 000045         #5-451
C$CLCK  = 000062         #5-451
C$CLEA  = 000012         #5-451      95-3395
C$CLOS  = 000035         #5-451
C$CLP1  = 000006         #5-451
C$CVEC  = 000036         #5-451      82-2935    82-2937    93-3345
C$DCLN  = 000044         #5-451      49-1826    91-3228    91-3248
C$DODU  = 000051         #5-451      72-2714    93-3357
C$DRPT  = 000024         #5-451      95-3374
C$DU    = 000053         #5-451      97-3433
```

SYMBOL CROSS REFERENCE
CREF

SYMBOL	VALUE	REFERENCES							
C$EDIT	= 000003	#5-451	5-494						
C$ERDF	= 000055	#5-451	72-2668	72-2682					
C$ERHR	= 000056	#5-451	72-2695	72-2710					
C$ERRO	= 000060	#5-451							
C$ERSF	= 000054	#5-451	49-1823	91-3227	91-3247				
C$ERSO	= 000057	#5-451	72-2688	72-2706	86-3019				
C$ESCA	= 000010	#5-451							
C$ESEG	= 000005	#5-451							
C$ESUB	= 000003	#5-451							
C$ETST	= 000001	#5-451	102-3552	104-3579	106-3621	108-3653	110-3684	112-3715	114-3746	116-3868
C$EXIT	= 000032	#5-451	102-3545	104-3559	106-3589	108-3629	110-3660	112-3691	114-3722	116-3758
C$GETB	= 000026	#5-451							
C$GETW	= 000027	#5-451							
C$GMAN	= 000043	#5-451							
C$GPHR	= 000042	#5-451	91-3234						
C$GPLO	= 000030	#5-451							
C$GPRI	= 000040	#5-451							
C$INIT	= 000011	#5-451	91-3315						
C$INLP	= 000020	#5-451							
C$MANI	= 000050	#5-451							
C$MEM	= 000031	#5-451							
C$MSG	= 000023	#5-451	74-2744						
C$OPEN	= 000034	#5-451							
C$PNTB	= 000014	#5-451	68-2504	74-2734	74-2737	74-2740	86-3020		
C$PNTF	= 000017	#5-451	88-3054	88-3059	88-3061	93-3354	97-3413	116-3848	116-3855	116-3857
		116-3859								
C$PNTS	= 000016	#5-451	89-3134	89-3136	89-3152	89-3153	89-3163			
C$PNTX	= 000015	#5-451	68-2480	68-2485	68-2488	68-2490	68-2511	68-2519		
C$QIO	= 000377	#5-451								
C$RDBU	= 000007	#5-451								
C$REFG	= 000047	#5-451	91-3207							
C$RESE	= 000033	#5-451	#5-451							
C$REVI	= 000003	#5-451	5-494							
C$RFLA	= 000021	#5-451	91-3285							
C$RPT	= 000025	#5-451	89-3176							
C$SEFG	= 000046	#5-451								
C$SPRI	= 000041	#5-451	82-2894							
C$SVEC	= 000037	#5-451	82-2895	82-2897	93-3334					
C$TPRI	= 000013	#5-451								
DESC	014574	86-3020	#86-3031							
DEVPTR	003310 G	#15-846	*49-1763	49-1764	49-1794	*49-1796	*49-1800	49-1801	49-1817	*49-1819
		*91-3210	91-3212	91-3219	*91-3221	*91-3230	91-3232	*91-3279	*116-3849	116-3850
		116-3860	*116-3862							
DEV0	003366	19-974	#19-986							
DEV1	003600	19-975	#19-987							
DEV2	004012	19-976	#19-988							
DEV3	004224	19-977	#19-989							
DEV4	004436	19-978	#19-990							
DEV5	004650	19-979	#19-991							
DEV6	005062	19-980	#19-992							
DEV7	005274	19-981	#19-993							
DFPTBL	002174 G	#8-539								

SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES								
DFTL1	012640		*72-2680	*72-2681	#72-2682	72-2682					
DIAGMC	= 000000		5-451	5-451							
DLV	= 000074	G	#16-911	56-2034	58-2120	58-2144	*62-2233	62-2234	62-2235	*62-2237	*62-2258
			62-2259	62-2260	*62-2262	66-2383	74-2738	74-2740	*74-2741	*84-2955	84-2956
			84-2957	*84-2959	*91-3276						
DOBRK	013654	G	49-1787	49-1807	#82-2873	116-3781					
DONE	003320	G	#15-850	*45-1712	47-1732	*102-3548	*104-3570	*106-3612	*108-3650	*110-3681	*112-3712
			*114-3743	*116-3842							
DR	= 000060	G	#16-904	37-1615	37-1617	*37-1619	*39-1647	39-1648	*39-1650	74-2734	76-2765
			*91-3239	91-3243	91-3245	104-3565	106-3596	106-3596	106-3597	106-3597	108-3638
			108-3639	108-3639	110-3669	110-3670	110-3670	112-3700	112-3701	112-3701	114-3731
			114-3732	114-3732							
DRVCHK	002216		#9-567	108-3636	110-3667	112-3698	114-3729				
EF.CON	= 000036	G	#10-640								
EF.NEW	= 000035	G	#10-640								
EF.PWR	= 000034	G	#10-640								
EF.RES	= 000037	G	#10-640								
EF.STA	= 000040	G	#10-640	91-3207							
ENDGP8	010114		58-2076	58-2160	#58-2162						
ENDRSP	007076		50-1875	#50-1927							
ENDT8	025460		116-3846	116-3861	#116-3866						
ERRDES	013034	G	72-2668	72-2682	72-2688	72-2695	72-2706	72-2710	#74-2727	86-3019	
ESABO	= 177720	G	#12-734								
ESCKS	= 177757	G	#12-745	12-747	12-748	12-749	70-2583				
ESCKSM	= 177757		#12-747								
ESCMD	= 177720	G	#12-743	70-2607							
ESNCRT	= 177767	G	#12-735	70-2602							
ESNOMO	= 177737	G	#12-742	70-2578							
ESNONX	= 177770	G	#12-736	70-2612							
ESOK	= 000000	G	#12-737	70-2566							
ESPART	= 177776	G	#12-738	70-2622							
ESRD	= 177757		#12-749								
ESREC	= 177711	G	#12-744	70-2627							
ES[illegible]	= 177740	G	#12-739	70-2596							
ESSLF	= 177777	G	#12-746								
ESTRY	= 000001	G	#12-740	70-2569							
ESWLOC	= 177765	G	#12-741	70-2617							
ESWR	= 177757		#12-748								
EVL	= 000004	G	#10-640	72-2676							
EVLTHR	002220		#9-568	72-2678							
EXOFF	007425		#56-2049	58-2151	62-2245						
EXON	007424		56-2005	#56-2048	58-2093	62-2253	62-2265				
E$END	= 002100		#5-451								
E$LOAD	= 000035		#5-451	5-494							
FLGLOC	016616	G	72-2676	*91-3285	#91-3324						
FM	015574		89-3153	89-3163	#89-3186						
FMO	015556		89-3152	#89-3183							
FTLNM	003316		#15-849	72-2699							
F$AU	= 000015		#5-451	99-3455	99-3478						
F$AUTO	= 000020		#5-451	93-3332	93-3346						
F$BGN	= 000040		#5-451	5-477	6-503	9-580	10-633	74-2727	84-2944	84-2952	88-3073
			89-3117	89-3124	89-3192	91-3202	93-3332	95-3369	97-3405	99-3455	102-3540

```
SYMBOL  VALUE          REFERENCES
                        102-3543  102-3545  102-3552  104-3557  104-3559  104-3579  106-3587  106-3589  106-3621
                        108-3627  108-3629  108-3653  110-3658  110-3660  110-3684  112-3689  112-3691  112-3715
                        114-3720  114-3722  114-3746  116-3751  116-3758  116-3868  120-3919  122-3984  122-3995
                        124-4038  126-4073  126-4075  126-4076  126-4076  126-4081  126-4082
F$CLEA  = 000007        #5-451    95-3369   95-3395
F$DU    = 000016        #5-451    97-3405   97-3433
F$END   = 000041        #5-451    5-451     5-451     5-451     5-451     5-451     5-451     5-451     5-451
                         5-451    5-451     5-451     5-451     5-451     5-451     5-451     5-451     5-477
                         9-580    10-633    74-2744   84-2948   84-2961   88-3073   89-3117   89-3176   89-3192
                        91-3315   93-3346   95-3395   97-3433   99-3478   102-3540  102-3543  102-3543  102-3543
                        102-3545  102-3552  102-3552  104-3557  104-3557  104-3557  104-3559  104-3579  104-3579
                        106-3587  106-3587  106-3587  106-3589  106-3621  106-3621  108-3627  108-3627  108-3627
                        108-3629  108-3653  108-3653  110-3658  110-3658  110-3658  110-3660  110-3684  110-3684
                        112-3689  112-3689  112-3689  112-3691  112-3715  112-3715  114-3720  114-3720  114-3720
                        114-3722  114-3746  114-3746  116-3751  116-3751  116-3751  116-3758  116-3868  116-3868
                        120-3919  122-3984  122-4010  124-4052  126-4073  126-4075  126-4076  126-4081  126-4082
F$HARD  = 000004        #5-451    122-3995  122-4010
F$HW    = 000013        #5-451    8-539     8-552
F$INIT  = 000006        #5-451    91-3202   91-3315
F$JMP   = 000050        #5-451    102-3545  104-3559  106-3589  108-3629  110-3660  112-3691  114-3722  116-3758
F$MOD   = 000000        #5-451    5-477     9-580     10-633    88-3073   89-3117   89-3192   102-3540  120-3919
                        122-3984  126-4073
F$MSG   = 000011        #5-451    74-2727   74-2744
F$PROT  = 000021        #5-451    6-503     6-507
F$PWR   = 000017        #5-451
F$RPT   = 000012        #5-451    89-3124   89-3176
F$SEG   = 000003        #5-451
F$SOFT  = 000005        #5-451    124-4038  124-4052
F$SRV   = 000010        #5-451    84-2944   84-2948   84-2952   84-2961
F$SUB   = 000002        #5-451
F$SW    = 000014        #5-451    9-561     9-578
F$TEST  = 000001        #5-451    102-3543  102-3552  104-3557  104-3579  106-3587  106-3621  108-3627  108-3653
                        110-3658  110-3684  112-3689  112-3715  114-3720  114-3746  116-3751  116-3868
GBTMP       010436      *62-2229  *62-2239  *62-2249   62-2250  *62-2269  #62-2275
GBTMP2      010440      *62-2244   62-2255   62-2257  *62-2264   62-2267  #62-2276
GETANS      007150  G    47-1734  #54-1971
GETHRD      016162       91-3226  #91-3230
GETPTR      007214      #54-1983
GETRS       017066       97-3409  #97-3434
GTAGIN      007462      #58-2078   58-2157
GTBYTE      010212  G    56-2013   56-2033   58-2097   58-2117   58-2141  #62-2229
GTDOWN      010052       58-2082   58-2086   58-2101   58-2112   58-2116   58-2119   58-2123   58-2143   58-2147
                         58-2149  #58-2153
GTOK        007720       58-2105  #58-2125
GTPKS1      007216  G    54-1980  #56-2000
GTPKS8      007426  G    54-1976  #58-2071   58-2161
GTPTR       010210      *58-2077   58-2078   58-2154  *58-2156  #60-2203
GTUM        007662       58-2110  #58-2114
G$CNTO  = 000200        #5-451
G$DELM  = 000372        #5-451
G$DISP  = 000003        #5-451
G$EXCP  = 000400        #5-451
```

```
SYMBOL  VALUE           REFERENCES
G$HILI  = 000002        #5-451
G$LOLI  = 000001        #5-451
G$NO    = 000000        #5-451
G$OFFS  = 000400        #5-451      122-3998    122-3999    122-4000    122-4001    122-4002    124-4039    124-4040    124-4041
                          124-4042    124-4043    124-4044    124-4045
G$OFSI  = 000376        #5-451      122-3998    122-3999    122-4000    122-4001    122-4002    124-4039    124-4040    124-4041
                          124-4042    124-4043    124-4044    124-4045
G$PRMA  = 000001        #5-451      122-3998    122-3999
G$PRMD  = 000002        #5-451      124-4039    124-4044
G$PRML  = 000000        #5-451      122-4000    122-4001    122-4002    124-4040    124-4041    124-4042    124-4043    124-4045
G$RADA  = 000140        #5-451
G$RADB  = 000000        #5-451
G$RADD  = 000040        #5-451      124-4039    124-4044
G$RADL  = 000120        #5-451      122-4000    122-4001    122-4002    124-4040    124-4041    124-4042    124-4043    124-4045
G$RADO  = 000020        #5-451      122-3998    122-3999
G$XFER  = 000004        #5-451
G$YES   = 000010        #5-451      122-3998    122-3999    122-4000    122-4001    122-4002    124-4039    124-4040    124-4041
                          124-4042    124-4043    124-4044    124-4045
HARDR   = 000136  G     #17-941     89-3148     89-3158
HARDW   = 000140  G     #17-942     89-3150     89-3160
HELP    = 000000        #5-436      5-446       5-468       5-486       7-509       7-524       8-546       9-571       #10-585
                          10-623      10-642      19-994      19-1000     21-1016     21-1021     21-1029     21-1036     21-1041
                          21-1047     37-1543     37-1555     37-1560     37-1566     37-1571     37-1577     37-1585     37-1592
                          37-1598     37-1604     #89-3079    91-3293     91-3303     95-3376     95-3383     97-3415     97-3421
                          99-3458     99-3464     #100-3484   100-3525    100-3531    120-3920    120-3925    120-3935    #122-3946
                          122-4004    122-4020    124-4046    126-4066
HOE     = 100000  G     #10-640
HRD       012772        72-2700     #72-2708
HRDRD   = 000016  G     #11-666     68-2522
HRDWR   = 000020  G     #11-667     68-2524
HRD1      011612        68-2509     #68-2518
IBE     = 010000  G     #10-640
IDPTR     003322  G     #15-851     *45-1713    45-1714     45-1716     *45-1718
IDU     = 000040  G     #10-640
IER     = 020000  G     #10-640
INIT      016030        #91-3204
INITWD    014406        *82-2873    82-2900     82-2908     82-2915     82-2922     82-2928     #84-2978
INIT2     016056        91-3208     #91-3210
ISR     = 000100  G     #10-640
IXE     = 004000  G     #10-640
I$AU    = 000041        #5-451      #99-3455    #99-3478
I$AUTO  = 000041        #5-451      #93-3332    #93-3346
I$CLN   = 000041        #5-451      #95-3369    #95-3395
I$DU    = 000041        #5-451      #97-3405    #97-3433
I$HRD   = 000041        #122-3995   #122-4010
I$INIT  = 000041        #5-451      #91-3202    #91-3315
I$MOD   = 000041        #5-451      5-477       #5-477      9-580       #9-580      10-633      #10-633     88-3073     #88-3073
                          89-3117     #89-3117    89-3192     #89-3192    102-3540    #102-3540   120-3919    #120-3919   122-3984
                          #122-3984   126-4073    #126-4073
I$MSG   = 000041        #5-451      #74-2727    #74-2744
I$PROT  = 000040        #5-451      #6-503
I$PTAB  = 000041        #5-451      126-4076    #126-4076   126-4081    #126-4081
```

SYMBOL CROSS REFERENCE CREF

```
SYMBOL  VALUE           REFERENCES
I$PWR   = 000041        #5-451
I$RPT   = 000041        #5-451      #89-3124    #89-3176
I$SEG   = 000041        #5-451       102-3543    104-3557    106-3587    108-3627    110-3658    112-3689    114-3720    116-3751
I$SETU  = 000041        #5-451       126-4075   #126-4075    126-4076    126-4082   #126-4082
I$SFT   = 000041        #124-4038   #124-4052
I$SRV   = 000041        #5-451      #84-2944    #84-2948    #84-2952    #84-2961
I$SUB   = 000041        #5-451       102-3543    104-3557    106-3587    108-3627    110-3658    112-3689    114-3720    116-3751
I$TST   = 000041        #5-451       102-3543   #102-3543    102-3545    102-3552   #102-3552   #102-3552    104-3557   #104-3557
                         104-3559    104-3579   #104-3579   #104-3579    106-3587   #106-3587    106-3589    106-3621   #106-3621
                        #106-3621    108-3627   #108-3627    108-3629    108-3653   #108-3653   #108-3653    110-3658   #110-3658
                         110-3660    110-3684   #110-3684   #110-3684    112-3689   #112-3689    112-3691    112-3715   #112-3715
                        #112-3715    114-3720   #114-3720    114-3722    114-3746   #114-3746   #114-3746    116-3751   #116-3751
                         116-3758    116-3868   #116-3868   #116-3868
J$JMP   = 000167        #5-451
LENGTH    002206        #9-563       91-3287
LGOFST  = 000120  G     #17-931      72-2660
LNCNT     014774        *88-3052    *88-3057    #88-3066
LOE     = 040000  G     #10-640
LOG       012500  G      52-1953     62-2273     66-2366     66-2386     66-2395     66-2423     68-2473     68-2496     68-2525
                         70-2633    #72-2652     82-2886     82-2930     84-2974
LOGO      013022         72-2689     72-2707    #72-2715
LOGOK     012566         72-2667    #72-2670
LOGOK2    012652         72-2677     72-2679    #72-2684
LOGO      012554        #72-2668
LOG1      012670        *72-2686    *72-2687    #72-2688     72-2688
LOG2      012720        *72-2693    *72-2694    #72-2695     72-2695
LOG3      012760        *72-2704    *72-2705    #72-2706     72-2706
LOG3B     013002        *72-2708    *72-2709    #72-2710     72-2710
LOT     = 000010  G     #10-640
LSTDEV    003364  G     #19-981      37-1621     39-1651     41-1673     45-1716     49-1794     49-1817     58-2154     64-2309
                         89-3164     91-3219     93-3341     116-3860
L$ACP     002110  G     #5-494
L$APT     002036  G     #5-494
L$AU      017146  G      5-494      #99-3455
L$AUT     002070  G     #5-494
L$AUTO    016620  G      5-494      #93-3332
L$CCP     002106  G     #5-494
L$CLEA    017002  G      5-494      #95-3369
L$CO      002032  G     #5-494
L$DEPO    002011  G     #5-494
L$DESC    002122  G      5-494      #5-496
L$DESP    002076  G     #5-494
L$DEVP    002060  G     #5-494
L$DISP    002152  G      5-494      #7-522
L$DLY     002116  G     #5-494
L$DTP     002040  G     #5-494
L$DTYP    002034  G     #5-494
L$DU      017022  G      5-494      #97-3405
L$DUT     002072  G     #5-494
L$DVTY    005506  G      5-494      #21-1014
L$EF      002052  G     #5-494
L$ENVI    002044  G     #5-494
```

SYMBOL	VALUE		REFERENCES				
L$ETP	002102	G	#5-494				
L$EXP1	002046	G	#5-494				
L$EXP4	002064	G	#5-494				
L$EXP5	002066	G	#5-494				
L$HARD	037614	G	5-494	122-3995	#122-3995		
L$HIME	002120	G	#5-494				
L$HPCP	002016	G	#5-494				
L$HPTP	002022	G	#5-494				
L$HW	002174	G	5-494	8-539	#8-539		
L$ICP	002104	G	#5-494				
L$INIT	016030	G	5-494	#91-3202			
L$LADP	002026	G	#5-494				
L$LAST	040422	G	5-494	#126-4072	126-4082		
L$LOAD	002100	G	#5-494				
L$LUN	002074	G	#5-494	*72-2657	*72-2658	*86-3017	*86-3018	*91-3233
L$MREV	002050	G	#5-494				
L$NAME	002000	G	#5-494				
L$PRIO	002042	G	#5-494				
L$PROT	002142	G	5-494	#6-503			
L$PRT	002112	G	#5-494				
L$REPP	002062	G	#5-494				
L$REV	002010	G	#5-494				
L$RPT	015014	G	5-494	#89-3124			
L$SOFT	037756	G	5-494	124-4038	#124-4038		
L$SPC	002056	G	#5-494				
L$SPCP	002020	G	#5-494				
L$SPTP	002024	G	#5-494				
L$STA	002030	G	#5-494				
L$SW	002206	G	5-494	9-561	#9-561		
L$TEST	002114	G	#5-494				
L$TIML	002014	G	#5-494				
L$UNIT	002012	G	#5-494	91-3225	91-3281		
L10001	002204		8-539	#8-552			
L10002	002224		9-561	#9-578			
L10003	013224		#74-2744				
L10004	014312		#84-2948				
L10005	014346		#84-2961				
L10006	015436		#89-3176				
L10007	016526		#91-3315				
L10010	016724		#93-3346				
L10011	017020		#95-3395				
L10012	017064		#97-3433				
L10013	017146		#99-3478				
L10014	017350		102-3545	#102-3552			
L10015	017622		104-3559	#104-3579			
L10016	021226		106-3589	#106-3621			
L10017	022216		108-3629	#108-3653			
L10020	023002		110-3660	#110-3684			
L10021	023772		112-3691	#112-3715			
L10022	024556		114-3722	#114-3746			
L10023	025462		116-3758	#116-3868			
L10024	037656		122-3995	#122-4010			

SYMBOL	VALUE		REFERENCES							
L10025	040040		124-4038	#124-4052						
L10026	040426		#126-4076							
L10030	040436		126-4076	#126-4081						
MABEE	012732		72-2692	#72-2698						
MODRSP	010116		58-2129	#58-2164						
MRSDLY	010120		*58-2132	*58-2136	#58-2166					
MRSP	= 000210	G	#17-963	50-1874	58-2074	64-2306	*66-2377	*66-2417	*91-3277	116-3853
MSAGE1	025464		116-3848	#116-3870						
MSAGE2	025525		116-3855	#116-3871						
MSAGE3	025561		116-3857	#116-3872						
MSAGE4	025610		116-3859	#116-3873						
MSAUTO	016762		93-3354	#93-3359						
MSBDA	002336	G	#14-788	86-3019						
MSCMD	002702	G	13-773	#14-810						
MSCOM	002402	G	13-763	#14-792						
MSG1	037656		122-3998	#122-4012						
MSG1B	037667		122-3999	#122-4013						
MSG1C	037704		122-4000	#122-4014						
MSG2	037722		122-4001	#122-4015						
MSG3	037737		122-4002	#122-4016						
MSG4	040040		124-4039	#124-4053						
MSG4B	040105		124-4040	#124-4054						
MSG5	040147		124-4041	#124-4055						
MSG6	040201		124-4042	#124-4056						
MSG7	040226		124-4043	#124-4057						
MSG8	040254		124-4044	#124-4058						
MSG9	040315		124-4045	#124-4059						
MSHCHK	002554	G	13-766	#14-802						
MSHDRD	003152	G	13-764	#14-826						
MSHDWR	003214	G	13-765	#14-828						
MSNIT	002616	G	13-770	#14-804						
MSNLOG	002320	G	13-757	13-778	#14-786					
MSNOMO	002444	G	13-769	#14-794						
MSNOTP	002462	G	13-779	#14-796						
MSNRSP	002762	G	13-777	#14-816						
MSOVRN	003256	G	13-762	#14-830						
MSPART	002632	G	13-771	#14-806						
MSQRSP	002776	G	13-761	#14-818						
MSREC	002716	G	13-774	#14-812						
MSRNIT	002534	G	13-760	#14-800						
MSSELF	002362	G	13-775	#14-790						
MSSFRD	003052	G	13-758	#14-822						
MSSFWR	003112	G	13-759	#14-824						
MSSKER	002304	G	13-767	#14-784						
MSTOSN	003030	G	13-780	#14-820						
MSUNIT	002654	G	13-772	#14-808						
MSWPRO	002512	G	13-768	#14-798						
MSWRSP	002736	G	13-776	#14-814						
MXRTRY	003326	G	#15-853	68-2508						
NCART	= 000054	G	#11-680	70-2604						
NODRVS	016560		91-3247	#91-3320						
NOMOR	006532		49-1823	#49-1830						

```
SYMBOL  VALUE          REFERENCES
NOMOT   = 000030  G    #11-671     70-2580
NOREE     011200       68-2459   #68-2463
NOUNIT  = 000036  G    #11-674     70-2614
NOXOFF    006630       #50-1876
NTSFT     012702       72-2685   #72-2691
NXTRET    006526       49-1793   49-1822   #49-1827
NXTST     006100  G    47-1730   #49-1762
NXTST2    006350       49-1795   #49-1799
ONEFIL  = 000001       #2-4      2-8       4-432     5-433     5-472     9-581     10-582    10-595    88-3075
                       89-3076   89-3089   99-3480   100-3481  100-3492  121-3941  122-3942  122-3956
OTL     = 000010  G    #11-663     66-2382
OVRFLO    013400       72-2668   #74-2749
OVRN    = 000012  G    #11-664     66-2385
O$APTS  = 000000       #5-451      5-494
O$AU    = 000001       #5-451    #5-484    5-494
O$BGNR  = 000001       #5-451    #5-484    5-494
O$BGNS  = 000001       #5-451    #5-484    5-494
O$DU    = 000001       #5-451    #5-484    5-494
O$ERRT  = 000000       #5-451      5-494
O$GNSW  = 000001       #5-451    #5-484    5-494
O$POIN  = 000001       #5-451    #5-484    #5-484    #5-484    #5-484    #5-484    #5-484    5-484     5-494
O$SETU  = 000001       #5-451    #5-484     5-494    126-4072
PARTL   = 000034  G    #11-673     70-2624
PATTEN  = 000072  G    #16-910     74-2736   86-3010   *91-3272  *106-3595  106-3596  *108-3635  *108-3638  108-3639
                       *110-3666 *110-3669 *112-3697 *112-3700  112-3701  *114-3728  *114-3731
PDTFLG    016614  G    *91-3241    91-3268   #91-3323
PERDEV    006362       #49-1801    49-1820
PKPTR   = 000104  G    #16-916   *50-1890    58-2096   58-2148   *58-2150  *66-2352  86-3002   88-3051
PNT     = 001000  G    #10-640
PPSOT8    002222       #9-569     116-3845
PRBUF     002212       #9-565      88-3049
PRDAT     014776       *88-3053    88-3054   #88-3067
PRFORM    015000        88-3054   #88-3068
PRI     = 002000  G    #10-640
PRI00   = 000000  G    #10-640     82-2894
PRI01   = 000040  G    #10-640
PRI02   = 000100  G    #10-640
PRI03   = 000140  G    #10-640
PRI04   = 000200  G    #10-640
PRI05   = 000240  G    #10-640
PRI06   = 000300  G    #10-640
PRI07   = 000340  G     5-494      #10-640    82-2895   82-2897   93-3334
PRNPAK    014630  G     86-3023   #88-3044
PRNSIZ    003334  G    #15-856    *86-3022  *88-3055
PTR       017116       *97-3434    97-3435  *97-3438  #97-3441
RCBCNT    003314       #15-848   *56-2011  *56-2022  *56-2027  *56-2031  *58-2090  *58-2109  *58-2113  *58-2115
                       *58-2139  *66-2354   66-2361   66-2428
RCBFSZ  = 001036  G    #12-718     43-1691   120-3904  120-3908  120-3909  120-3910  120-3911  120-3912  120-3913
                       120-3914  120-3915
RCDB    = 000024  G    #16-896     62-2233   62-2258   82-2893   84-2955  *91-3264
RCFLG     003312  G    #15-847   *56-2012   56-2018  *58-2088   58-2104  *66-2355
RCINIT  = 000006  G    #11-662     66-2394
```

```
SYMBOL  VALUE              REFERENCES
RCSR    = 000022  G        #16-895    62-2231    62-2247    82-2917    82-2934    84-2954    *91-3262   93-3339
RCVBUF  = 000102  G        #16-915    43-1690    50-1890    56-2008    66-2346    *91-3217
RCVHND    014314           #84-2954
RCVINT    014314  G         82-2895   #84-2952
RDN0    = 000114  G        #16-920    *50-1908   89-3153
RDN1    = 000116  G        #16-921    *50-1910   89-3163
REC     = 000064  G        #16-906     50-1886   74-2735   *91-3274   *104-3563   104-3565   *106-3594   106-3596   106-3596
                            106-3597   106-3597  *106-3603   106-3604  *106-3607  *106-3608  *108-3532   108-3635   108-3639
                            108-3639  *108-3642  *108-3647  *110-3663   110-3666   110-3670   110-3670   *110-3673  *110-3678
                           *112-3694   112-3697   112-3701   112-3701  *112-3704  *112-3709  *114-3725   114-3728   114-3732
                            114-3732  *114-3735  *114-3740   116-3792   116-3819
RECDAT    017606           104-3563   104-3568  #104-3573
RECERR  = 000042  G        #11-676    70-2629
RECID     013306  G         74-2737   #74-2747
RECID2    013462            74-2740   #74-2751
RECOV     011710            68-2485   #68-2536
RETERR    012112            68-2504   #68-2546
RETRY   = 000002  G        #16-887    *68-2479   68-2480    68-2485   *68-2497   *68-2505   68-2508    *68-2510   68-2511
                           *68-2526   *91-3273
RLUN      015440           *89-3142   *89-3143   89-3152   #89-3177
RPTR      015442           *89-3133    89-3138   89-3164   *89-3166  #89-3178
RSCMND  = 000002  G        #12-698     12-704    50-1896    102-3547   104-3565   104-3565   106-3596   106-3596   106-3597
                            106-3597   108-3639  108-3639    110-3670   110-3670   112-3701   112-3701   114-3732   114-3732
                            116-3762   116-3789  116-3805    116-3816   116-3832
RSCONT  = 000020  G        #12-699     58-2164    66-2358    82-2922    106-3596   106-3596   108-3639   108-3639   112-3701
                            112-3701
RSDASZ  = 000204  G        #12-711     12-713     12-718    56-2027    66-2399    66-2406
RSDATA  = 000001  G        #12-703     56-2025    58-2111    58-2148    66-2373    66-2397    106-3596   106-3596   106-3597
                            106-3597   108-3639  108-3639    110-3670   110-3670   112-3701   112-3701   114-3732   114-3732
                            116-3775
RSDNSZ  = 000222  G        #12-713     58-2113
RSEND   = 000002  G        #12-704     56-2020    58-2107    66-2368    66-2409    102-3547   106-3596   106-3596   106-3597
                            106-3597   108-3639  108-3639    110-3670   110-3670   112-3701   112-3701   114-3732   114-3732
RSGCDP  = 000034  G        #12-715    116-3776
RSINIT  = 000004  G        #12-702     66-2391    84-2978
RSMSIZ  = 000012  G        #12-709     12-717    102-3547   102-3547   104-3565   104-3565   106-3596   106-3596   106-3596
                            106-3596   106-3597  106-3597    106-3597   106-3597   108-3639   108-3639   108-3639   108-3639
                            110-3670   110-3670  110-3670    110-3670   112-3701   112-3701   112-3701   112-3701   114-3732
                            114-3732   114-3732  114-3732    116-3763   116-3770   116-3790   116-3798   116-3817   116-3825
RSNDSZ  = 000016  G        #12-707     12-713     12-718    56-2022    58-2109    102-3547   104-3565   106-3596   106-3596
                            106-3597   106-3597  108-3639    108-3639   110-3670   110-3670   112-3701   112-3701   114-3732
                            114-3732   116-3806  116-3833
RSNTAB    002224           #13-757    72-2673
RSSEND  = 000100  G        #12-723
RSSGET  = 000012  G        #12-727    116-3764
RSSNIT  = 000001  G        #12-729     82-2876   116-3791
RSSNOP  = 000000  G        #12-728    116-3818
RSSNSZ  = 000016  G        #12-717     66-2411   102-3547   104-3565   106-3596   106-3596   106-3597   106-3597   108-3639
                            108-3639   110-3670  110-3670    112-3701   112-3701   114-3732   114-3732   116-3772   116-3804
                            116-3831
RSSRD   = 000002  G        #12-725     50-1903    70-2571   106-3596   106-3597   106-3597   108-3639   110-3670   110-3670
                            112-3701   114-3732  114-3732
```

```
SYMBOL  VALUE          REFERENCES
RSSSEK  = 000005  G    #12-726    104-3565
RSSSLF  = 000007  G    #12-730    70-2588    102-3547
RSSWR   = 000003  G    #12-724    50-1912    70-2575    106-3596    108-3639    112-3701
RSVP      006556  G    #50-1867   102-3547   104-3565   106-3596    106-3596    106-3596    106-3597    106-3597    108-3639
                       108-3639   108-3639   110-3670   110-3670    112-3701    112-3701    112-3701    114-3732    114-3732
                       116-3779   116-3809   116-3836
RSXOFF  = 000023  G    #12-701    56-2049    120-3901   120-3902
RSXON   = 000020  G    #12-700    56-2048
RTRTN     012050       68-2480    68-2511    #68-2542
RUN       006050  G    #47-1730   47-1739    102-3544   102-3544    104-3558    104-3558    106-3588    106-3588    108-3628
                       108-3628   110-3659   110-3659   112-3690    112-3690    114-3721    114-3721    116-3755
SAVCNT  = 000206  G    #17-962    49-1774    *49-1785
SECREC    003332  G    #15-855    *91-3289   *91-3292   108-3647    110-3678    112-3709    114-3740
SERVST    010206       *54-1974   58-2158    *60-2188   #60-2202
SETDR     005630  G    #39-1645   102-3544   104-3558   106-3588    108-3628    110-3659    112-3690    114-3721
SETLEN    016470       #91-3287
SETPTR    005704       *39-1645   39-1646    39-1651    *39-1653    #39-1656
SETSRV    010122       58-2081    58-2085    #60-2178
SETUP     006002  G    #45-1712   102-3544   102-3544   104-3558    104-3558    106-3588    106-3588    108-3628    108-3628
                       110-3659   110-3659   112-3690   112-3690    114-3721    114-3721    116-3754
SFPTBL    002206  G    #9-561
SFT       012750       72-2701    #72-2704
SFTOUT    040040       #124-4052
SFTRD   = 000002  G    #11-660    68-2493
SFTWR   = 000004  G    #11-661    68-2495
SKERR   = 000024  G    #11-669    70-2598
SLFER   = 000044  G    #11-677    70-2593
SND       006646       50-1877    #50-1880   50-1884
SNDBYT    007100  G    50-1880    #52-1942   56-2006    58-2095     58-2130     58-2152     62-2246     62-2254     62-2266
SNDCNT  = 000070  G    #16-909    *50-1879   *50-1883   *102-3547   *104-3565   *106-3596   *106-3596   *106-3596   *106-3596
                       *106-3597  *106-3597  *108-3639  *108-3639   *108-3639   *108-3639   *110-3670   *110-3670   *112-3701
                       *112-3701  *112-3701  *112-3701  *114-3732   *114-3732   *116-3772   *116-3804   *116-3831
SNDHND    014300       #84-2946
SNDINT    014300  G    82-2897    #84-2944
SOFTR   = 000122  G    #17-932    89-3144    89-3154
SOFTW   = 000124  G    #17-933    89-3146    89-3156
SRVTBL    010166       60-2182    #60-2193
STAEOP    002210       #9-564     95-3372
STATHD    015444       89-3134    #89-3180
STATUS  = 000000  G    #16-886    116-3851
STHD2     015720       89-3136    #89-3189
STRT      016612  G    *91-3205   *91-3209   91-3250    #91-3322
SUCCS   = 000076  G    #16-912    *50-1888   68-2467    68-2483     *70-2565    74-2737     *91-3275
SUCOTL  = 000046  G    #11-678    70-2632
SVCGBL  = 000000       #5-451     #5-460     5-494      5-494       5-494       5-494       5-494       5-494       5-494
                       5-494      5-494      5-494      5-494       5-494       5-494       5-494       5-494       5-494
                       5-494      5-494      5-494      5-494       5-494       5-494       5-494       5-494       5-494
                       5-494      5-494      5-494      5-494       5-494       5-494       5-494       5-494       5-494
                       5-494      5-494      5-494      5-494       5-496       6-503       7-522       8-539       8-539
                       9-561      9-561      21-1014    74-2727     84-2944     84-2952     89-3124     91-3202     93-3332
                       95-3369    97-3405    99-3455    122-3995    124-4038    #126-4072   126-4072    126-4072    126-4072
SVCINS  = 000001       #5-451     #5-457     5-494      5-494       5-494       5-494       5-494       5-494       5-494
```

```
SYMBOL  VALUE               REFERENCES
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
                             5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-496
                             5-496      5-496      5-496      5-496      5-496      7-522      7-522      7-522      7-522
                             7-522      7-522      7-522      7-522      7-522      7-522      7-522      7-522      7-522
                             7-522      7-522      7-522      7-522      7-522      7-522      7-522      7-522      7-522
                             7-522      7-522      7-522      7-522      7-522      8-539      8-539      8-539      9-561
                             9-561      9-561      21-1014    21-1014    21-1014    21-1014    21-1014    21-1014    47-1736
                             47-1736    47-1736    49-1823    49-1823    49-1823    49-1823    49-1823    49-1823    49-1823
                             49-1823    49-1823    49-1823    49-1823    49-1823    49-1824    49-1824    49-1824    49-1826
                             49-1826    49-1826    52-1947    52-1947    52-1947    62-2252    62-2252    62-2252    68-2480
                             68-2480    68-2480    68-2480    68-2480    68-2480    68-2480    68-2480    68-2480    68-2480
                             68-2480    68-2480    68-2480    68-2480    68-2480    68-2480    68-2480    68-2480    68-2485
                             68-2485    68-2485    68-2485    68-2485    68-2485    68-2485    68-2485    68-2485    68-2485
                             68-2485    68-2485    68-2485    68-2485    68-2485    68-2485    68-2485    68-2485    68-2488
                             68-2488    68-2488    68-2488    68-2488    68-2488    68-2488    68-2488    68-2488    68-2488
                             68-2488    68-2488    68-2488    68-2488    68-2488    68-2490    68-2490    68-2490    68-2490
                             68-2490    68-2490    68-2490    68-2490    68-2490    68-2490    68-2490    68-2490    68-2490
                             68-2490    68-2490    68-2504    68-2504    68-2504    68-2504    68-2504    68-2504    68-2504
                             68-2504    68-2504    68-2504    68-2504    68-2504    68-2504    68-2504    68-2504    68-2511
                             68-2511    68-2511    68-2511    68-2511    68-2511    68-2511    68-2511    68-2511    68-2511
                             68-2511    68-2511    68-2511    68-2511    68-2511    68-2511    68-2511    68-2511    68-2519
                             68-2519    68-2519    68-2519    68-2519    68-2519    68-2519    68-2519    68-2519    68-2519
                             68-2519    68-2519    68-2519    68-2519    68-2519    72-2668    72-2668    72-2668    72-2668
                             72-2668    72-2668    72-2668    72-2668    72-2668    72-2668    72-2668    72-2668    72-2682
                             72-2682    72-2682    72-2682    72-2682    72-2682    72-2682    72-2682    72-2682    72-2682
                             72-2682    72-2682    72-2688    72-2688    72-2688    72-2688    72-2688    72-2688    72-2688
                             72-2688    72-2688    72-2688    72-2688    72-2688    72-2695    72-2695    72-2695    72-2695
                             72-2695    72-2695    72-2695    72-2695    72-2695    72-2695    72-2695    72-2695    72-2706
                             72-2706    72-2706    72-2706    72-2706    72-2706    72-2706    72-2706    72-2706    72-2706
                             72-2706    72-2706    72-2710    72-2710    72-2710    72-2710    72-2710    72-2710    72-2710
                             72-2710    72-2710    72-2710    72-2710    72-2710    72-2714    72-2714    72-2714    74-2734
                             74-2734    74-2734    74-2734    74-2734    74-2734    74-2734    74-2734    74-2734    74-2734
                             74-2734    74-2734    74-2734    74-2734    74-2734    74-2734    74-2734    74-2734    74-2734
                             74-2734    74-2734    74-2734    74-2734    74-2734    74-2734    74-2734    74-2734    74-2734
                             74-2734    74-2734    74-2737    74-2737    74-2737    74-2737    74-2737    74-2737    74-2737
                             74-2737    74-2737    74-2737    74-2737    74-2737    74-2737    74-2737    74-2737    74-2737
                             74-2737    74-2737    74-2737    74-2737    74-2737    74-2737    74-2737    74-2737    74-2737
                             74-2737    74-2737    74-2737    74-2737    74-2737    74-2737    74-2737    74-2737    74-2737
                             74-2737    74-2737    74-2740    74-2740    74-2740    74-2740    74-2740    74-2740    74-2740
```

SYMBOL	VALUE	REFERENCES								
		74-2740	74-2740	74-2740	74-2740	74-2740	74-2740	74-2740	74-2740	74-2740
		74-2740	74-2740	74-2744	74-2744	74-2744	82-2879	82-2879	82-2879	82-2894
		82-2894	82-2894	82-2894	82-2894	82-2894	82-2895	82-2895	82-2895	82-2895
		82-2895	82-2895	82-2895	82-2895	82-2895	82-2895	82-2895	82-2895	82-2895
		82-2895	82-2895	82-2895	82-2895	82-2895	82-2897	82-2897	82-2897	82-2897
		82-2897	82-2897	82-2897	82-2897	82-2897	82-2897	82-2897	82-2897	82-2897
		82-2897	82-2897	82-2897	82-2897	82-2897	82-2935	82-2935	82-2935	82-2935
		82-2935	82-2935	82-2937	82-2937	82-2937	82-2937	82-2937	82-2937	84-2948
		84-2948	84-2948	84-2961	84-2961	84-2961	84-2969	84-2969	84-2969	84-2970
		84-2970	84-2970	86-3019	86-3019	86-3019	86-3019	86-3019	86-3019	86-3019
		86-3019	86-3019	86-3019	86-3019	86-3019	86-3020	86-3020	86-3020	86-3020
		86-3020	86-3020	86-3020	86-3020	86-3020	86-3020	86-3020	86-3020	86-3020
		86-3020	86-3020	86-3020	86-3020	86-3020	88-3054	88-3054	88-3054	88-3054
		88-3054	88-3054	88-3054	88-3054	88-3054	88-3054	88-3054	88-3054	88-3054
		88-3054	88-3054	88-3054	88-3054	88-3054	88-3054	88-3054	88-3054	88-3059
		88-3059	88-3059	88-3059	88-3059	88-3059	88-3059	88-3059	88-3059	88-3059
		88-3059	88-3059	88-3059	88-3059	88-3059	88-3061	88-3061	88-3061	88-3061
		88-3061	88-3061	88-3061	88-3061	88-3061	88-3061	88-3061	88-3061	88-3061
		88-3061	88-3061	89-3132	89-3132	89-3132	89-3134	89-3134	89-3134	89-3134
		89-3134	89-3134	89-3134	89-3134	89-3134	89-3134	89-3134	89-3134	89-3134
		89-3134	89-3134	89-3135	89-3135	89-3135	89-3136	89-3136	89-3136	89-3136
		89-3136	89-3136	89-3136	89-3136	89-3136	89-3136	89-3136	89-3136	89-3136
		89-3136	89-3136	89-3137	89-3137	89-3137	89-3152	89-3152	89-3152	89-3152
		89-3152	89-3152	89-3152	89-3152	89-3152	89-3152	89-3152	89-3152	89-3152
		89-3152	89-3152	89-3152	89-3152	89-3152	89-3153	89-3153	89-3153	89-3153
		89-3153	89-3153	89-3153	89-3153	89-3153	89-3153	89-3153	89-3153	89-3153
		89-3153	89-3153	89-3153	89-3153	89-3153	89-3153	89-3153	89-3153	89-3153
		89-3153	89-3153	89-3153	89-3153	89-3153	89-3153	89-3153	89-3153	89-3153
		89-3153	89-3153	89-3153	89-3153	89-3153	89-3153	89-3153	89-3153	89-3153
		89-3153	89-3153	89-3163	89-3163	89-3163	89-3163	89-3163	89-3163	89-3163
		89-3163	89-3163	89-3163	89-3163	89-3163	89-3163	89-3163	89-3163	89-3163
		89-3163	89-3163	89-3163	89-3163	89-3163	89-3163	89-3163	89-3163	89-3163
		89-3163	89-3163	89-3163	89-3163	89-3163	89-3163	89-3163	89-3163	89-3163
		89-3163	89-3163	89-3163	89-3163	89-3163	89-3163	89-3163	89-3163	89-3176
		89-3176	89-3176	91-3207	91-3207	91-3207	91-3207	91-3207	91-3207	91-3208
		91-3208	91-3208	91-3227	91-3227	91-3227	91-3227	91-3227	91-3227	91-3227
		91-3227	91-3227	91-3227	91-3227	91-3227	91-3228	91-3228	91-3228	91-3234
		91-3234	91-3234	91-3234	91-3234	91-3234	91-3234	91-3234	91-3234	91-3235
		91-3235	91-3235	91-3247	91-3247	91-3247	91-3247	91-3247	91-3247	91-3247
		91-3247	91-3247	91-3247	91-3247	91-3247	91-3248	91-3248	91-3248	91-3285
		91-3285	91-3285	91-3285	91-3285	91-3285	91-3315	91-3315	91-3315	93-3334
		93-3334	93-3334	93-3334	93-3334	93-3334	93-3334	93-3334	93-3334	93-3334
		93-3334	93-3334	93-3334	93-3334	93-3334	93-3334	93-3334	93-3334	93-3345
		93-3345	93-3345	93-3345	93-3345	93-3345	93-3346	93-3346	93-3346	93-3354
		93-3354	93-3354	93-3354	93-3354	93-3354	93-3354	93-3354	93-3354	93-3354
		93-3354	93-3354	93-3354	93-3354	93-3354	93-3357	93-3357	93-3357	95-3374
		95-3374	95-3374	95-3395	95-3395	95-3395	97-3413	97-3413	97-3413	97-3413
		97-3413	97-3413	97-3413	97-3413	97-3413	97-3413	97-3413	97-3413	97-3413
		97-3413	97-3413	97-3413	97-3413	97-3413	97-3433	97-3433	97-3433	99-3478
		99-3478	99-3478	102-3545	102-3545	102-3545	102-3545	102-3545	102-3545	102-3552
		102-3552	102-3552	104-3559	104-3559	104-3559	104-3559	104-3559	104-3559	104-3579
		104-3579	104-3579	106-3589	106-3589	106-3589	106-3589	106-3589	106-3589	106-3621

SYMBOL CROSS REFERENCE CREF

SYMBOL		VALUE		REFERENCES								
				106-3621	106-3621	108-3629	108-3629	108-3629	108-3629	108-3629	108-3629	108-3653
				108-3653	108-3653	110-3660	110-3660	110-3660	110-3660	110-3660	110-3660	110-3684
				110-3684	110-3684	112-3691	112-3691	112-3691	112-3691	112-3691	112-3691	112-3715
				112-3715	112-3715	114-3722	114-3722	114-3722	114-3722	114-3722	114-3722	114-3746
				114-3746	114-3746	116-3758	116-3758	116-3758	116-3758	116-3758	116-3758	116-3848
				116-3848	116-3848	116-3848	116-3848	116-3848	116-3848	116-3848	116-3848	116-3848
				116-3848	116-3848	116-3848	116-3848	116-3848	116-3855	116-3855	116-3855	116-3855
				116-3855	116-3855	116-3855	116-3855	116-3855	116-3855	116-3855	116-3855	116-3855
				116-3855	116-3855	116-3855	116-3855	116-3855	116-3857	116-3857	116-3857	116-3857
				116-3857	116-3857	116-3857	116-3857	116-3857	116-3857	116-3857	116-3857	116-3857
				116-3857	116-3857	116-3857	116-3857	116-3857	116-3859	116-3859	116-3859	116-3859
				116-3859	116-3859	116-3859	116-3859	116-3859	116-3859	116-3859	116-3859	116-3859
				116-3859	116-3859	116-3859	116-3859	116-3859	116-3868	116-3868	116-3868	122-3995
				122-3995	122-3995	122-3998	122-3998	122-3998	122-3998	122-3998	122-3998	122-3998
				122-3998	122-3998	122-3998	122-3998	122-3998	122-3999	122-3999	122-3999	122-3999
				122-3999	122-3999	122-3999	122-3999	122-3999	122-3999	122-3999	122-3999	122-4000
				122-4000	122-4000	122-4000	122-4000	122-4000	122-4000	122-4000	122-4000	122-4001
				122-4001	122-4001	122-4001	122-4001	122-4001	122-4001	122-4001	122-4001	122-4002
				122-4002	122-4002	122-4002	122-4002	122-4002	122-4002	122-4002	122-4002	122-4010
				122-4010	122-4010	124-4038	124-4038	124-4038	124-4039	124-4039	124-4039	124-4039
				124-4039	124-4039	124-4039	124-4039	124-4039	124-4039	124-4039	124-4039	124-4039
				124-4039	124-4039	124-4040	124-4040	124-4040	124-4040	124-4040	124-4040	124-4040
				124-4040	124-4040	124-4041	124-4041	124-4041	124-4041	124-4041	124-4041	124-4041
				124-4041	124-4041	124-4042	124-4042	124-4042	124-4042	124-4042	124-4042	124-4042
				124-4042	124-4042	124-4043	124-4043	124-4043	124-4043	124-4043	124-4043	124-4043
				124-4043	124-4043	124-4044	124-4044	124-4044	124-4044	124-4044	124-4044	124-4044
				124-4044	124-4044	124-4044	124-4044	124-4044	124-4044	124-4044	124-4044	124-4045
				124-4045	124-4045	124-4045	124-4045	124-4045	124-4045	124-4045	124-4045	124-4052
				124-4052	124-4052	126-4072	126-4072	126-4072	126-4072	126-4072	126-4072	126-4072
				126-4072	126-4072	126-4076	126-4076	126-4076	126-4076	126-4076	126-4076	
SVCSUB	=	000001		#5-451	#5-459							
SVCTAG	=	000001		#5-451	#5-461	8-552	9-578	74-2744	84-2948	84-2961	89-3176	91-3315
				93-3346	95-3395	97-3433	99-3478	102-3552	104-3579	106-3621	108-3653	110-3684
				112-3715	114-3746	116-3868	122-4010	124-4052	126-4076	126-4081		
SVCTST	=	000001		#5-451	#5-458	102-3543	104-3557	106-3587	108-3627	110-3658	112-3689	114-3720
				116-3751								
SWAPDR		005526	G	#37-1610	102-3544	104-3558	106-3588	108-3628	110-3659	112-3690	114-3721	
SWPTR		005626		*37-1611	37-1612	37-1621	*37-1623	#37-1631				
SYSTAT		003304	G	#15-836	*49-1776	*49-1781	*49-1790	54-1972	*56-2017	*58-2103	64-2294	*66-2353
				68-2458	68-2460	*68-2529	74-2734	*78-2786	*78-2789	78-2800	*82-2928	*91-3284
S$LSYM	=	010000		#5-451	#8-552	#9-578	#74-2744	#84-2948	#84-2961	#89-3176	#91-3315	#93-3346
				#95-3395	#97-3433	#99-3478	#102-3552	#104-3579	#106-3621	#108-3653	#110-3684	#112-3715
				#114-3746	#116-3868	#122-4010	#124-4052					
TAPLEN		003306	G	#15-845	*91-3287	*91-3288	91-3290	108-3633	108-3648	110-3664	110-3679	112-3695
				112-3710	114-3726	114-3741						
TEST8		003340	G	#15-858	50-1872	58-2072	58-2091	58-2126	64-2304	66-2375	66-2415	*91-3206
				*116-3760	*116-3786	*116-3843						
THRSHI		012016		68-2490	#68-2540							
THRSLO		011770		68-2488	#68-2538							
TMP	=	000066	G	#16-908	*106-3592	106-3608	*106-3609	*108-3633	*108-3640	*108-3648	*110-3664	*110-3671
				*110-3679	*112-3695	*112-3702	*112-3710	*114-3726	*114-3733	*114-3741		
TOMANY		016530		91-3227	#91-3318							

```
SYMBOL  VALUE          REFERENCES
TORCVB  = 000050  G    #11-679     62-2272     84-2973
TOSNDB  = 000056  G    #11-681     52-1952     82-2885
TRBUF     026174        50-1876     50-1878     50-1885    102-3547    104-3565    106-3596    106-3596    106-3596    106-3596
                        106-3597    106-3597    108-3639    108-3639    108-3639    108-3639    110-3670    110-3670    112-3701
                        112-3701    112-3701    112-3701    114-3732    114-3732    116-3761    116-3788    116-3815   #120-3904
TRK     = 000062  G    #16-905    *108-3634    108-3644   *108-3646   *110-3665    110-3675   *110-3677   *112-3696    112-3706
                       *112-3708   *114-3727    114-3737   *114-3739
TRPHND    016730        93-3334    #93-3354
TRPPTR    016726       *93-3335     93-3336     93-3341    *93-3343    #93-3347
TSTPC   = 000020  G    #16-894     *45-1715     49-1792     49-1813    *50-1868
TSTTOP    003324       #15-852      45-1715    *102-3544   *104-3558   *106-3588   *108-3628   *110-3659   *112-3690   *114-3721
                       *116-3753
TST1      017214        102-3544   #102-3547
TST2      017416        104-3558   #104-3562
TST3      017670        106-3588   #106-3592
TST3PT    021214        106-3595    106-3599   #106-3614
TST4      021274        108-3628   #108-3632
TST4EX    022210        108-3645   #108-3650
TST5      022264        110-3659   #110-3663
TST5EX    022774        110-3676   #110-3681
TST6      023050        112-3690   #112-3694
TST6EX    023764        112-3707   #112-3712
TST7      024040        114-3721   #114-3725
TST7EX    024550        114-3738   #114-3743
TST8      024602        116-3753   #116-3760
TUVECT  = 000204  G    #17-961      82-2895    *82-2896     82-2897    *82-2898     82-2935    *82-2936     82-2937    *82-2938
                       *91-3238
T$ARGC  = 000002       #5-494       5-494      #5-494       5-494       5-494      #5-494       5-494       5-494      #5-494
                        5-494       5-494      #5-494       5-494       5-494      #5-494       5-494       5-494      #68-2480
                        68-2480    #68-2480     68-2480     68-2480    #68-2485     68-2485    #68-2485     68-2485     68-2485
                       #68-2488     68-2488     68-2488    #68-2490     68-2490     68-2490    #68-2504     68-2504     68-2504
                       #68-2511     68-2511    #68-2511     68-2511     68-2511    #68-2519     68-2519     68-2519    #74-2734
                        74-2734    #74-2734     74-2734    #74-2734     74-2734    #74-2734     74-2734     74-2734    #74-2737
                        74-2737    #74-2737     74-2737    #74-2737     74-2737    #74-2737     74-2737    #74-2737     74-2737
                        74-2737    #74-2740     74-2740    #74-2740     74-2740     74-2740    #86-3020     86-3020    #86-3020
                        86-3020     86-3020    #88-3054     88-3054    #88-3054     88-3054     88-3054    #88-3059     88-3059
                        88-3059    #88-3061     88-3061     88-3061    #89-3134     89-3134     89-3134    #89-3136     89-3136
                        89-3136    #89-3152     89-3152    #89-3152     89-3152     89-3152    #89-3153     89-3153    #89-3153
                        89-3153    #89-3153     89-3153    #89-3153     89-3153    #89-3153     89-3153    #89-3153     89-3153
                       #89-3153     89-3153    #89-3153     89-3153    #89-3153     89-3153     89-3153    #89-3163     89-3163
                       #89-3163     89-3163    #89-3163     89-3163    #89-3163     89-3163    #89-3163     89-3163    #89-3163
                        89-3163    #89-3163     89-3163    #89-3163     89-3163    #89-3163     89-3163     89-3163    #93-3354
                        93-3354     93-3354    #97-3413     97-3413    #97-3413     97-3413     97-3413    #116-3848    116-3848
                        116-3848   #116-3855    116-3855   #116-3855    116-3855    116-3855   #116-3857    116-3857   #116-3857
                        116-3857    116-3857   #116-3859    116-3859   #116-3859    116-3859    116-3859
T$CODE  = 006130       #122-3998    122-3998   #122-3998    122-3998   #122-3998    122-3998   #122-3999    122-3999   #122-3999
                        122-3999   #122-3999    122-3999   #122-4000    122-4000   #122-4000    122-4000   #122-4000    122-4000
                       #122-4001    122-4001   #122-4001    122-4001   #122-4001    122-4001   #122-4002    122-4002   #122-4002
                        122-4002   #122-4002    122-4002   #124-4039    124-4039   #124-4039    124-4039   #124-4039    124-4039
                       #124-4040    124-4040   #124-4040    124-4040   #124-4040    124-4040   #124-4041    124-4041   #124-4041
                        124-4041   #124-4041    124-4041   #124-4042    124-4042   #124-4042    124-4042   #124-4042    124-4042
                       #124-4043    124-4043   #124-4043    124-4043   #124-4043    124-4043   #124-4044    124-4044   #124-4044
```

SYMBOL VALUE REFERENCES

T$ERRN = 000146
124-4044 #124-4044 124-4044 #124-4045 124-4045 #124-4045 124-4045 #124-4045 124-4045
#5-451 #49-1823 49-1823 #72-2668 72-2668 #72-2682 72-2682 #72-2688 72-2688
#72-2695 72-2695 #72-2706 72-2706 #72-2710 72-2710 #86-3019 86-3019 #91-3227
91-3227 #91-3247 91-3247

T$EXCP = 000000
T$FLAG = 000040
#122-3998 122-3998 #122-3999 122-3999 #124-4039 124-4039 #124-4044 124-4044
#102-3545 #102-3545 102-3545 102-3545 #104-3559 #104-3559 104-3559 104-3559 #106-3589
#106-3589 106-3589 106-3589 #108-3629 #108-3629 108-3629 108-3629 #110-3660 #110-3660
110-3660 110-3660 #112-3691 #112-3691 112-3691 112-3691 #114-3722 #114-3722 114-3722
114-3722 #116-3758 #116-3758 116-3758 116-3758

T$FREE = 040436
126-4072 #126-4082

T$GMAN = 000000
#5-451

T$HILI = 000376
#122-3998 122-3998 #122-3999 122-3999 #124-4039 124-4039 #124-4044 124-4044

T$LAST = 000001
#5-451 #126-4072 126-4075

T$LOLI = 000001
#122-3998 122-3998 #122-3999 122-3999 #124-4039 124-4039 #124-4044 124-4044

T$LSYM = 010000
#5-451 5-451 8-552 9-578 74-2744 84-2948 84-2961 89-3176 91-3315
93-3346 95-3395 97-3433 99-3478 102-3552 104-3579 106-3621 108-3653 110-3684
112-3715 114-3746 116-3868 122-4010 124-4052

T$LTNO = 000010
#126-4072

T$NEST = 177777
#5-451 5-477 #5-477 5-477 6-503 #6-503 6-503 6-507 6-507
6-507 #6-507 8-539 #8-539 8-539 8-552 8-552 8-552 #8-552
9-561 #9-561 9-561 9-578 9-578 9-578 #9-578 9-580 9-580
9-580 #9-580 10-633 #10-633 10-633 74-2727 #74-2727 74-2727 74-2744
74-2744 74-2744 #74-2744 84-2944 #84-2944 84-2944 84-2948 84-2948 84-2948
#84-2948 84-2952 #84-2952 84-2952 84-2961 84-2961 84-2961 #84-2961 88-3073
88-3073 88-3073 #88-3073 89-3117 #89-3117 89-3117 89-3124 #89-3124 89-3124
89-3176 89-3176 89-3176 #89-3176 89-3192 89-3192 89-3192 #89-3192 91-3202
#91-3202 91-3202 91-3315 91-3315 91-3315 #91-3315 93-3332 #93-3332 93-3332
93-3346 93-3346 93-3346 #93-3346 95-3369 #95-3369 95-3369 95-3395 95-3395
95-3395 #95-3395 97-3405 #97-3405 97-3405 97-3433 97-3433 97-3433 #97-3433
99-3455 #99-3455 99-3455 99-3478 99-3478 99-3478 #99-3478 102-3540 #102-3540
102-3540 102-3543 #102-3543 102-3543 102-3552 102-3552 102-3552 #102-3552 104-3557
#104-3557 104-3557 104-3579 104-3579 104-3579 #104-3579 106-3587 #106-3587 106-3587
106-3621 106-3621 106-3621 #106-3621 108-3627 #108-3627 108-3627 108-3653 108-3653
108-3653 #108-3653 110-3658 #110-3658 110-3658 110-3684 110-3684 110-3684 #110-3684
112-3689 #112-3689 112-3689 112-3715 112-3715 112-3715 #112-3715 114-3720 #114-3720
114-3720 114-3746 114-3746 114-3746 #114-3746 116-3751 #116-3751 116-3751 116-3868
116-3868 116-3868 #116-3868 120-3919 120-3919 120-3919 #120-3919 122-3984 #122-3984
122-3984 122-3995 #122-3995 122-3995 122-4010 122-4010 122-4010 #122-4010 124-4038
#124-4038 124-4038 124-4052 124-4052 124-4052 #124-4052 126-4073 126-4073 126-4073
#126-4073

T$NS0 = 000000
#5-477 9-580 #10-633 88-3073 #89-3117 89-3192 #91-3202 91-3315 #93-3332
93-3346 #95-3369 95-3395 #97-3405 97-3433 #99-3455 99-3478 #102-3540 120-3919
#122-3984 126-4073

T$NS1 = 000005
#6-503 6-507 #8-539 8-552 #9-561 9-578 #74-2727 74-2744 #84-2944
84-2948 #84-2952 84-2961 #89-3124 89-3176 #102-3543 102-3552 #104-3557 104-3579
#106-3587 106-3621 #108-3627 108-3653 #110-3658 110-3684 #112-3689 112-3715 #114-3720
114-3746 #116-3751 116-3868 #122-3995 122-4010 #124-4038 124-4052

T$PCNT = 000000
#126-4075 126-4076 #126-4076 126-4076

T$PTAB = 010027
#126-4076 126-4076

T$PTHV = 000001
5-494 #126-4082

T$PTNU = 000001
#5-451 126-4076 #126-4076 126-4082 126-4082

T$SAVL = 177777
#5-451

SYMBOL	VALUE	REFERENCES								
T$SEGL	= 177777	#5-451								
T$SIZE	= 000006	126-4072	#126-4082							
T$SUBN	= 000000	#5-451	#102-3543	#104-3557	#106-3587	#108-3627	#110-3658	#112-3689	#114-3720	#116-3751
T$TAGL	= 177777	#5-451								
T$TAGN	= 010031	#5-451	6-503	6-503	#6-503	8-539	8-539	#8-539	9-561	9-561
		#9-561	74-2727	74-2727	#74-2727	84-2944	84-2944	#84-2944	84-2952	84-2952
		#84-2952	89-3124	89-3124	#89-3124	91-3202	91-3202	#91-3202	93-3332	93-3332
		#93-3332	95-3369	95-3369	#95-3369	97-3405	97-3405	#97-3405	99-3455	99-3455
		#99-3455	102-3543	102-3543	#102-3543	104-3557	104-3557	#104-3557	106-3587	106-3587
		#106-3587	108-3627	108-3627	#108-3627	110-3658	110-3658	#110-3658	112-3689	112-3689
		#112-3689	114-3720	114-3720	#114-3720	116-3751	116-3751	#116-3751	122-3995	122-3995
		#122-3995	124-4038	124-4038	#124-4038	126-4075	126-4075	#126-4075	126-4076	126-4076
		#126-4076	126-4076	126-4076	#126-4076					
T$TEMP	= 000000	#6-507	6-507	#7-522	7-522	7-522	#7-522	7-522	7-522	#7-522
		7-522	7-522	#7-522	7-522	7-522	#7-522	7-522	7-522	#7-522
		7-522	7-522	#7-522	7-522	7-522	#7-522	7-522	7-522	#7-522
		#8-552	8-552	#9-578	9-578	#9-580	9-580	#74-2744	74-2744	#84-2948
		84-2948	#84-2961	84-2961	#88-3073	88-3073	#89-3176	89-3176	#89-3192	89-3192
		#91-3315	91-3315	#93-3346	93-3346	#95-3395	95-3395	#97-3433	97-3433	#99-3478
		99-3478	#102-3545	102-3545	#102-3552	102-3552	#104-3559	104-3559	#104-3579	104-3579
		#106-3589	106-3589	#106-3621	106-3621	#108-3629	108-3629	#108-3653	108-3653	#110-3660
		110-3660	#110-3684	110-3684	#112-3691	112-3691	#112-3715	112-3715	#114-3722	114-3722
		#114-3746	114-3746	#116-3758	116-3758	#116-3868	116-3868	#120-3919	120-3919	#122-3998
		122-3998	#122-3998	122-3998	#122-3998	122-3998	#122-3999	122-3999	#122-3999	122-3999
		#122-3999	122-3999	#122-4000	122-4000	#122-4000	122-4000	#122-4000	122-4000	#122-4001
		122-4001	#122-4001	122-4001	#122-4001	122-4001	#122-4002	122-4002	#122-4002	122-4002
		#122-4002	122-4002	#122-4010	122-4010	#124-4039	124-4039	#124-4039	124-4039	#124-4039
		124-4039	#124-4040	124-4040	#124-4040	124-4040	#124-4040	124-4040	#124-4041	124-4041
		#124-4041	124-4041	#124-4041	124-4041	#124-4042	124-4042	#124-4042	124-4042	#124-4042
		124-4042	#124-4043	124-4043	#124-4043	124-4043	#124-4043	124-4043	#124-4044	124-4044
		#124-4044	124-4044	#124-4044	124-4044	#124-4045	124-4045	#124-4045	124-4045	#124-4045
		124-4045	#124-4052	124-4052	#126-4073	126-4073				
T$TEST	= 000010	#5-451	102-3543	#102-3543	102-3543	104-3557	#104-3557	104-3557	106-3587	#106-3587
		106-3587	108-3627	#108-3627	108-3627	110-3658	#110-3658	110-3658	112-3689	#112-3689
		112-3689	114-3720	#114-3720	114-3720	116-3751	#116-3751	116-3751	126-4072	
T$TSTM	= 177777	#5-451	47-1736	49-1823	49-1824	49-1826	52-1947	62-2252	68-2480	68-2485
		68-2488	68-2490	68-2504	68-2511	68-2519	72-2668	72-2682	72-2688	72-2695
		72-2706	72-2710	72-2714	74-2734	74-2737	74-2740	74-2744	82-2879	82-2894
		82-2895	82-2897	82-2935	82-2937	84-2969	84-2970	86-3019	86-3020	88-3054
		88-3059	88-3061	89-3132	89-3134	89-3135	89-3136	89-3137	89-3152	89-3153
		89-3163	89-3176	91-3207	91-3227	91-3228	91-3234	91-3247	91-3248	91-3285
		91-3315	93-3334	93-3345	93-3346	93-3354	93-3357	95-3374	95-3395	97-3413
		97-3433	99-3478	102-3545	102-3552	104-3559	104-3579	106-3589	106-3621	108-3629
		108-3653	110-3660	110-3684	112-3691	112-3715	114-3722	114-3746	116-3758	116-3848
		116-3855	116-3857	116-3859	116-3868					
T$TSTS	= 000001	#5-451	#102-3543	#104-3557	#106-3587	#108-3627	#110-3658	#112-3689	#114-3720	#116-3751
T$$AU	= 010013	#99-3455	99-3478							
T$$AUT	= 010010	#93-3332	93-3346							
T$$CLE	= 010011	#95-3369	95-3395							
T$$DAT	= 010030	#126-4076	126-4076	126-4081						
T$$DU	= 010012	#97-3405	97-3433							
T$$HAR	= 010024	#122-3995	122-3995	122-4010						

SYMBOL	VALUE		REFERENCES								
T$$HW	= 010001		#8-539	8-539	8-552						
T$$INI	= 010007		#91-3202	91-3315							
T$$MSG	= 010003		#74-2727	74-2744							
T$$PC	= 000001		#126-4075	126-4082							
T$$PRO	= 010000		#6-503								
T$$PTA	= 010027		#126-4075	126-4076	#126-4076						
T$$RPT	= 010006		#89-3124	89-3176							
T$$SOF	= 010025		#124-4038	124-4038	124-4052						
T$$SRV	= 010005		#84-2944	84-2948	#84-2952	84-2961					
T$$SW	= 010002		#9-561	9-561	9-578						
T$$TES	= 010023		#102-3543	102-3545	102-3552	#104-3557	104-3559	104-3579	#106-3587	106-3589	106-3621
			#108-3627	108-3629	108-3653	#110-3658	110-3660	110-3684	#112-3689	112-3691	112-3715
			#114-3720	114-3722	114-3746	#116-3751	116-3758	116-3868			
T1	017150	G	7-522	#102-3543							
T1TRY	= 000146	G	#17-945								
T2	017352	G	7-522	#104-3557							
T3	017624	G	7-522	#106-3587							
T4	021230	G	7-522	#108-3627							
T4TRY	= 000132	G	#17-939								
T5	022220	G	7-522	#110-3658							
T6	023004	G	7-522	#112-3689							
T7	023774	G	7-522	#114-3720							
T8	024560	G	7-522	#116-3751							
UAM	= 000200	G	#10-640								
UNIT	013226	G	74-2734	#74-2745							
UNITNO	025640		*116-3847	116-3855	116-3857	116-3859	*116-3863	#116-3875			
UNREC	012070		68-2519	#68-2544							
UNSUC	011466		68-2484	#68-2501							
UNXPCT	007624		#58-2106								
WAIT	014350		82-2903	82-2911	82-2918	#84-2965	84-2972				
WHCHDR	013504	G	50-1906	50-1920	72-2663	#76-2763					
WRLOCK	= 000026	G	#11-670	70-2619	72-2691						
WRTN0	= 000110	G	#16-918	*50-1922	89-3153	91-3254	91-3256				
WRTN1	= 000112	G	#16-919	*50-1925	89-3163						
XFNSND	006636		50-1873	#50-1878							
XMDB	= 000030	G	#16-898	52-1955	82-2888	84-2947	*91-3271				
XMSR	= 000026	G	#16-897	52-1944	82-2875	82-2881	82-2892	82-2902	82-2910	82-2933	84-2946
			*91-3266								
XSCNT	= 000036	G	#16-901	*102-3547	*104-3565	*106-3596	*106-3596	*106-3596	*108-3639	*108-3639	*108-3639
			*112-3701	*112-3701	*112-3701	*116-3776	*116-3806	*116-3833			
XSFLG	= 000034	G	#16-900	50-1891	56-2001	66-2348	*102-3547	*104-3565	*106-3596	*106-3596	*106-3596
			106-3596	106-3597	106-3597	*108-3639	*108-3639	*108-3639	108-3639	110-3670	110-3670
			*112-3701	*112-3701	*112-3701	112-3701	114-3732	114-3732	*116-3775	*116-3805	*116-3832
XSPKNM	= 000032	G	#16-899	50-1887	*50-1887	56-2009	58-2083	*58-2087	*58-2100	*58-2106	*58-2122
			*58-2146	66-2347	*102-3547	*104-3565	*106-3596	*106-3596	*106-3596	*106-3596	*106-3597
			*106-3597	*108-3639	*108-3639	*108-3639	*108-3639	*110-3670	*110-3670	*112-3701	*112-3701
			*112-3701	*112-3701	*114-3732	*114-3732	*116-3777	*116-3807	*116-3834		
XSPTR	= 000106	G	#16-917	*50-1893	58-2088	*58-2089	58-2090	*58-2153			
X$ALWA	= 000000		#5-451								
X$FALS	= 000040		#5-451								
X$OFFS	= 000400		#5-451								
X$TRUE	= 000020		#5-451								

MACRO NAME	REFERENCES									
BGNAU	#99-3455									
BGNAUT	93-3332									
BGNCLN	#95-3369									
BGNDU	#97-3405									
BGNHRD	122-3995									
BGNHW	#8-539									
BGNINI	91-3202									
BGNMOD	5-477	10-633	89-3117	102-3540	122-3984					
BGNMSG	#74-2727									
BGNPRO	6-503									
BGNPTA	#126-4076									
BGNRPT	89-3124									
BGNSET	#126-4075									
BGNSFT	#124-4038									
BGNSRV	84-2944	84-2952								
BGNSW	#9-561									
BGNTST	102-3543	104-3557	106-3587	108-3627	110-3658	112-3689	114-3720	116-3751		
BNCOMP	91-3208	91-3235								
BREAK	47-1736	49-1824	52-1947	62-2252	82-2879	84-2969	84-2970	89-3132	89-3135	89-3137
CLRVEC	82-2935	82-2937	93-3345							
DESCRI	#5-496									
DEVTYP	#21-1014									
DISPAT	7-522									
DOCLN	#49-1826	#91-3228	#91-3248							
DODU	72-2714	93-3357								
DORPT	95-3374									
ENDAU	99-3478									
ENDAUT	#93-3346									
ENDCLN	95-3395									
ENDDU	97-3433									
ENDHRD	#122-4010									
ENDHW	8-552									
ENDINI	#91-3315									
ENDMOD	9-580	88-3073	89-3192	120-3919	126-4073					
ENDMSG	74-2744									
ENDPRO	#6-507									
ENDPTA	126-4081									
ENDRPT	#89-3176									
ENDSET	#126-4082									
ENDSFT	124-4052									
ENDSRV	#84-2948	#84-2961								
ENDSW	9-578									
ENDTST	102-3552	104-3579	106-3621	108-3653	110-3684	112-3715	114-3746	116-3868		
EQUALS	10-640									
ERRDF	72-2668	72-2682								
ERRHRD	#72-2695	#72-2710								
ERRSF	#49-1823	#91-3227	#91-3247							
ERRSOF	72-2688	72-2706	86-3019							
EXIT	#102-3545	#104-3559	#106-3589	#108-3629	#110-3660	#112-3691	#114-3722	#116-3758		
GPHARD	91-3234									
GPRMA	#122-3998	#122-3999								
GPRMD	124-4039	124-4044								

```
MACRO NAME    REFERENCES
GPRML         #122-4000  #122-4001  #122-4002  #124-4040  #124-4041  #124-4042  #124-4043  #124-4045
HEADER         5-494
LASTAD         126-4072
M$BYTE        #5-494     #5-494     #5-494     #5-494
M$CHEC        #102-3545   102-3545  #104-3559   104-3559  #106-3589   106-3589  #108-3629   108-3629  #110-3660   110-3660
              #112-3691   112-3691  #114-3722   114-3722  #116-3758   116-3758
M$CNTO        #122-3998   122-3998  #122-3999   122-3999  #122-4000   122-4000  #122-4001   122-4001  #122-4002   122-4002
              #124-4039   124-4039  #124-4040   124-4040  #124-4041   124-4041  #124-4042   124-4042  #124-4043   124-4043
              #124-4044   124-4044  #124-4045   124-4045
M$COUN        #68-2480    68-2480   #68-2485    68-2485   #68-2488    68-2488   #68-2490    68-2490   #68-2504    68-2504
              #68-2511    68-2511   #68-2519    68-2519   #74-2734    74-2734    74-2734    74-2734   #74-2737    74-2737
               74-2737    74-2737    74-2737   #74-2740    74-2740   #86-3020    86-3020   #88-3054    88-3054   #88-3059
               88-3059   #88-3061    88-3061   #89-3134    89-3134   #89-3136    89-3136   #89-3152    89-3152   #89-3153
               89-3153    89-3153    89-3153    89-3153    89-3153    89-3153    89-3153    89-3153   #89-3163    89-3163
               89-3163    89-3163    89-3163    89-3163    89-3163    89-3163    89-3163   #93-3354    93-3354   #97-3413
               97-3413   #116-3848   116-3848  #116-3855   116-3855  #116-3857   116-3857  #116-3859   116-3859
M$DATA        #5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
               5-494      5-494      5-494      5-494      5-494      5-494     #5-494      5-494      5-494      5-494
               5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494
               5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494      5-494     #5-496
               5-496     #21-1014    21-1014
M$DECR        #6-507      6-507     #8-552      8-552     #9-578      9-578     #9-580      9-580     #74-2744    74-2744
              #84-2948    84-2948   #84-2961    84-2961   #88-3073    88-3073   #89-3176    89-3176   #89-3192    89-3192
              #91-3315    91-3315   #93-3346    93-3346   #95-3395    95-3395   #97-3433    97-3433   #99-3478    99-3478
              #102-3552   102-3552  #104-3579   104-3579  #106-3621   106-3621  #108-3653   108-3653  #110-3684   110-3684
              #112-3715   112-3715  #114-3746   114-3746  #116-3868   116-3868  #120-3919   120-3919  #122-4010   122-4010
              #124-4052   124-4052  #126-4073   126-4073  #126-4076   126-4076
M$DEFA        #122-3998  #122-3998  #122-3999  #122-3999  #122-4000  #122-4000  #122-4001  #122-4001  #122-4002  #122-4002
              #124-4039  #124-4039  #124-4040  #124-4040  #124-4041  #124-4041  #124-4042  #124-4042  #124-4043  #124-4043
              #124-4044  #124-4044  #124-4045  #124-4045
M$ENDE        #8-552     #9-578     #9-580     #74-2744   #84-2948   #84-2961   #88-3073   #89-3176   #89-3192   #91-3315
              #93-3346   #95-3395   #97-3433   #99-3478   #102-3552  #104-3579  #106-3621  #108-3653  #110-3684  #112-3715
              #114-3746  #116-3868  #120-3919  #122-4010  #124-4052  #126-4073
M$ERRI        #49-1823   #49-1823   #72-2668   #72-2668   #72-2682   #72-2682   #72-2688   #72-2688   #72-2695   #72-2695
              #72-2706   #72-2706   #72-2710   #72-2710   #86-3019   #86-3019   #91-3227   #91-3227   #91-3247   #91-3247
M$EXCP        #122-3998   122-3998   122-3998  #122-3999   122-3999   122-3999  #124-4039   124-4039   124-4039  #124-4044
               124-4044   124-4044
M$EXIT        #102-3545   102-3545  #104-3559   104-3559  #106-3589   106-3589  #108-3629   108-3629  #110-3660   110-3660
              #112-3691   112-3691  #114-3722   114-3722  #116-3758   116-3758
M$EXSE        #102-3545  #104-3559  #106-3589  #108-3629  #110-3660  #112-3691  #114-3722  #116-3758
M$EXTJ        #102-3545  #104-3559  #106-3589  #108-3629  #110-3660  #112-3691  #114-3722  #116-3758
M$GEN         #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494
              #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494
              #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494
              #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494
              #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494
              #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494
              #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-494
              #5-494     #5-494     #5-494     #5-494     #5-494     #5-494     #5-496     #5-496     #6-503     #6-503
              #7-522     #7-522     #8-539     #8-539     #8-539     #8-539     #8-552     #8-552     #9-561     #9-561
              #9-561     #9-561     #9-578     #9-578     #21-1014   #21-1014   #74-2727   #74-2727   #74-2744   #74-2744
              #84-2944   #84-2944   #84-2948   #84-2948   #84-2952   #84-2952   #84-2961   #84-2961   #89-3124   #89-3124
```

```
MACRO NAME   REFERENCES
             #89-3176  #89-3176  #91-3202  #91-3202  #91-3315  #91-3315  #93-3332  #93-3332  #93-3346  #93-3346
             #95-3369  #95-3369  #95-3395  #95-3395  #97-3405  #97-3405  #97-3433  #97-3433  #99-3455  #99-3455
             #99-3478  #99-3478  #102-3543 #102-3543 #102-3552 #102-3552 #104-3557 #104-3557 #104-3579 #104-3579
             #106-3587 #106-3587 #106-3621 #106-3621 #108-3627 #108-3627 #108-3653 #108-3653 #110-3658 #110-3658
             #110-3684 #110-3684 #112-3689 #112-3689 #112-3715 #112-3715 #114-3720 #114-3720 #114-3746 #114-3746
             #116-3751 #116-3751 #116-3868 #116-3868 #122-3995 #122-3995 #122-4010 #122-4010 #124-4038 #124-4038
             #124-4052 #124-4052 #126-4072 #126-4072 #126-4076 #126-4076 #126-4081 #126-4081
MSGETS       #6-507     6-507    #8-552     8-552    #9-578     9-578    #9-580     9-580    #74-2744   74-2744
             #84-2948   84-2948  #84-2961   84-2961  #88-3073   88-3073  #89-3176   89-3176  #89-3192   89-3192
             #91-3315   91-3315  #93-3346   93-3346  #95-3395   95-3395  #97-3433   97-3433  #99-3478   99-3478
             #102-3552  102-3552 #104-3579  104-3579 #106-3621  106-3621 #108-3653  108-3653 #110-3684  110-3684
             #112-3715  112-3715 #114-3746  114-3746 #116-3868  116-3868 #120-3919  120-3919 #122-4010  122-4010
             #124-4052  124-4052 #126-4073  126-4073
MSGETT       #102-3545 #104-3559 #106-3589 #108-3629 #110-3660 #112-3691 #114-3722 #116-3758
MSGNGB       #5-477    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494
             #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494
             #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494
             #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494
             #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494
             #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494
             #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494
             #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-494    #5-496    #5-496    #6-503
             #6-503    #7-522    #7-522    #8-539    #8-539    #8-539    #9-561    #9-561    #9-561    #10-633
             #21-1014  #21-1014  #74-2727  #74-2727  #84-2944  #84-2944  #84-2952  #84-2952  #89-3117  #89-3124
             #89-3124  #91-3202  #91-3202  #93-3332  #93-3332  #95-3369  #95-3369  #97-3405  #97-3405  #99-3455
             #99-3455  #102-3540 #122-3984 #122-3995 #122-3995 #124-4038 #124-4038 #126-4072 #126-4072
MSGNIN       #5-494     5-494     5-494     5-494     5-494     5-494    #5-494     5-494    #5-494     5-494
             #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494
             #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494
             #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494
             #5-494     5-494    #5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494     5-494
             #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494
             #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494
             #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494
             #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494    #5-494     5-494
             #5-496    #5-496     5-496     5-496    #7-522     7-522    #7-522     7-522    #7-522     7-522
             #7-522     7-522    #7-522     7-522    #7-522     7-522    #7-522     7-522    #7-522     7-522
             #7-522     7-522    #8-539     8-539    #9-561     9-561    #21-1014  #21-1014   21-1014   21-1014
             #47-1736   47-1736  #49-1823  #49-1823   49-1823  #49-1823   49-1823  #49-1823   49-1823  #49-1823
              49-1823  #49-1824   49-1824  #49-1826   49-1826  #52-1947   52-1947  #62-2252   62-2252  #68-2480
             #68-2480   68-2480  #68-2480   68-2480  #68-2480   68-2480   68-2480  #68-2480   68-2480   68-2480
             #68-2485  #68-2485   68-2485  #68-2485   68-2485  #68-2485   68-2485   68-2485  #68-2485   68-2485
              68-2485  #68-2488  #68-2488   68-2488  #68-2488   68-2488   68-2488  #68-2488   68-2488   68-2488
             #68-2490  #68-2490   68-2490  #68-2490   68-2490   68-2490  #68-2490   68-2490   68-2490  #68-2504
             #68-2504   68-2504  #68-2504   68-2504   68-2504  #68-2504   68-2504   68-2504  #68-2511  #68-2511
              68-2511  #68-2511   68-2511  #68-2511   68-2511   68-2511  #68-2511   68-2511   68-2511  #68-2519
             #68-2519   68-2519  #68-2519   68-2519   68-2519  #68-2519   68-2519   68-2519  #72-2668  #72-2668
              72-2668  #72-2668   72-2668  #72-2668   72-2668  #72-2668   72-2668  #72-2682  #72-2682   72-2682
             #72-2682   72-2682  #72-2682   72-2682  #72-2682   72-2682  #72-2688  #72-2688   72-2688  #72-2688
              72-2688  #72-2688   72-2688  #72-2688   72-2688  #72-2695  #72-2695   72-2695  #72-2695   72-2695
             #72-2695   72-2695  #72-2695   72-2695  #72-2706  #72-2706   72-2706  #72-2706   72-2706  #72-2706
              72-2706  #72-2706   72-2706  #72-2710  #72-2710   72-2710  #72-2710   72-2710  #72-2710   72-2710
```

MACRO NAME	REFERENCES									
	#72-2710	72-2710	#72-2714	#72-2714	72-2714	#74-2734	#74-2734	74-2734	74-2734	#74-2734
	74-2734	#74-2734	74-2734	74-2734	#74-2734	74-2734	#74-2734	74-2734	74-2734	#74-2734
	74-2734	74-2734	#74-2737	#74-2737	74-2737	74-2737	#74-2737	74-2737	74-2737	#74-2737
	74-2737	74-2737	#74-2737	74-2737	#74-2737	74-2737	#74-2737	74-2737	74-2737	#74-2737
	74-2737	74-2737	#74-2740	#74-2740	74-2740	#74-2740	74-2740	#74-2740	74-2740	74-2740
	#74-2740	74-2740	74-2740	#74-2744	74-2744	#82-2879	82-2879	#82-2894	82-2894	#82-2894
	82-2894	#82-2895	#82-2895	82-2895	#82-2895	82-2895	#82-2895	82-2895	#82-2895	82-2895
	#82-2895	82-2895	82-2895	#82-2897	#82-2897	82-2897	#82-2897	82-2897	#82-2897	82-2897
	#82-2897	82-2897	#82-2897	82-2897	82-2897	#82-2935	82-2935	#82-2935	82-2935	#82-2937
	82-2937	#82-2937	82-2937	#84-2948	84-2948	#84-2961	84-2961	#84-2969	84-2969	#84-2970
	84-2970	#86-3019	#86-3019	86-3019	#86-3019	86-3019	#86-3019	86-3019	#86-3019	86-3019
	#86-3020	#86-3020	86-3020	#86-3020	86-3020	#86-3020	86-3020	86-3020	#86-3020	86-3020
	86-3020	#88-3054	#88-3054	88-3054	88-3054	#88-3054	88-3054	#88-3054	88-3054	88-3054
	#88-3054	88-3054	88-3054	#88-3059	#88-3059	88-3059	#88-3059	88-3059	88-3059	#88-3059
	88-3059	88-3059	#88-3061	#88-3061	88-3061	#88-3061	88-3061	88-3061	#88-3061	88-3061
	88-3061	#89-3132	89-3132	#89-3134	#89-3134	89-3134	#89-3134	89-3134	89-3134	#89-3134
	89-3134	89-3134	#89-3135	89-3135	#89-3136	#89-3136	89-3136	#89-3136	89-3136	89-3136
	#89-3136	89-3136	89-3136	#89-3137	89-3137	#89-3152	#89-3152	89-3152	#89-3152	89-3152
	#89-3152	89-3152	89-3152	#89-3152	89-3152	89-3152	#89-3153	#89-3153	89-3153	#89-3153
	89-3153	#89-3153	89-3153	#89-3153	89-3153	#89-3153	89-3153	89-3153	#89-3153	89-3153
	#89-3153	89-3153	#89-3153	89-3153	#89-3153	89-3153	#89-3153	89-3153	89-3153	#89-3153
	89-3153	89-3153	#89-3163	#89-3163	89-3163	#89-3163	89-3163	#89-3163	89-3163	#89-3163
	89-3163	#89-3163	89-3163	89-3163	#89-3163	89-3163	#89-3163	89-3163	#89-3163	89-3163
	#89-3163	89-3163	#89-3163	89-3163	89-3163	#89-3163	89-3163	89-3163	#89-3176	89-3176
	#91-3207	91-3207	#91-3207	91-3207	#91-3208	91-3208	#91-3227	#91-3227	91-3227	#91-3227
	91-3227	#91-3227	91-3227	#91-3227	91-3227	#91-3228	91-3228	#91-3234	91-3234	#91-3234
	91-3234	#91-3234	91-3234	#91-3235	91-3235	#91-3247	#91-3247	91-3247	#91-3247	91-3247
	#91-3247	91-3247	#91-3247	91-3247	#91-3248	91-3248	#91-3285	91-3285	#91-3285	91-3285
	#91-3315	91-3315	#93-3334	#93-3334	93-3334	#93-3334	93-3334	#93-3334	93-3334	#93-3334
	93-3334	#93-3334	93-3334	93-3334	#93-3345	93-3345	#93-3345	93-3345	#93-3346	93-3346
	#93-3354	#93-3354	93-3354	#93-3354	93-3354	93-3354	#93-3354	93-3354	93-3354	#93-3357
	#93-3357	93-3357	#95-3374	95-3374	#95-3395	95-3395	#97-3413	#97-3413	97-3413	#97-3413
	97-3413	#97-3413	97-3413	97-3413	#97-3413	97-3413	97-3413	#97-3433	97-3433	#99-3478
	99-3478	#102-3545	102-3545	#102-3545	102-3545	#102-3552	102-3552	#104-3559	104-3559	#104-3559
	104-3559	#104-3579	104-3579	#106-3589	106-3589	#106-3589	106-3589	#106-3621	106-3621	#108-3629
	108-3629	#108-3629	108-3629	#108-3653	108-3653	#110-3660	110-3660	#110-3660	110-3660	#110-3684
	110-3684	#112-3691	112-3691	#112-3691	112-3691	#112-3715	112-3715	#114-3722	114-3722	#114-3722
	114-3722	#114-3746	114-3746	#116-3758	116-3758	#116-3758	116-3758	#116-3848	#116-3848	116-3848
	#116-3848	116-3848	116-3848	#116-3848	116-3848	116-3848	#116-3855	#116-3855	116-3855	#116-3855
	116-3855	#116-3855	116-3855	116-3855	#116-3855	116-3855	116-3855	#116-3857	#116-3857	116-3857
	#116-3857	116-3857	#116-3857	116-3857	116-3857	#116-3857	116-3857	116-3857	#116-3859	#116-3859
	116-3859	#116-3859	116-3859	#116-3859	116-3859	116-3859	#116-3859	116-3859	116-3859	#116-3868
	116-3868	#122-3995	122-3995	#122-3998	122-3998	122-3998	122-3998	122-3998	#122-3999	122-3999
	122-3999	122-3999	122-3999	#122-4000	122-4000	122-4000	122-4000	#122-4001	122-4001	122-4001
	122-4001	#122-4002	122-4002	122-4002	122-4002	#122-4010	122-4010	#124-4038	124-4038	#124-4039
	124-4039	124-4039	124-4039	124-4039	124-4039	#124-4040	124-4040	124-4040	124-4040	#124-4041
	124-4041	124-4041	124-4041	#124-4042	124-4042	124-4042	124-4042	#124-4043	124-4043	124-4043
	124-4043	#124-4044	124-4044	124-4044	124-4044	124-4044	124-4044	#124-4045	124-4045	124-4045
	124-4045	#124-4052	124-4052	#126-4072	126-4072	126-4072	126-4072	#126-4076	#126-4076	126-4076
	126-4076									
MSGNTA	#8-552	#8-552	#9-578	#9-578	#74-2744	#74-2744	#84-2948	#84-2948	#84-2961	#84-2961
	#89-3176	#89-3176	#91-3315	#91-3315	#93-3346	#93-3346	#95-3395	#95-3395	#97-3433	#97-3433

MACRO NAME	REFERENCES									
	#99-3478	#99-3478	#102-3552	#102-3552	#104-3579	#104-3579	#106-3621	#106-3621	#108-3653	#108-3653
	#110-3684	#110-3684	#112-3715	#112-3715	#114-3746	#114-3746	#116-3868	#116-3868	#122-4010	#122-4010
	#124-4052	#124-4052	#126-4076	#126-4076	#126-4081	#126-4081				
MSGNTE	#102-3543	#102-3543	#104-3557	#104-3557	#106-3587	#106-3587	#108-3627	#108-3627	#110-3658	#110-3658
	#112-3689	#112-3689	#114-3720	#114-3720	#116-3751	#116-3751				
MSHAPT	#5-494	#5-494								
MSHNAP	#5-494	#5-494								
MSINCR	#5-477	#5-477	#6-503	#6-503	#6-503	#6-503	#8-539	#8-539	#8-539	#8-539
	#9-561	#9-561	#9-561	#9-561	#10-633	#10-633	#47-1736	#49-1823	#49-1824	#49-1826
	#52-1947	#62-2252	#68-2480	#68-2485	#68-2488	#68-2490	#68-2504	#68-2511	#68-2519	#72-2668
	#72-2682	#72-2688	#72-2695	#72-2706	#72-2710	#72-2714	#74-2727	#74-2727	#74-2727	#74-2727
	#74-2734	#74-2737	#74-2740	#74-2744	#82-2879	#82-2894	#82-2895	#82-2897	#82-2935	#82-2937
	#84-2944	#84-2944	#84-2944	#84-2944	#84-2952	#84-2952	#84-2952	#84-2952	#84-2969	#84-2970
	#86-3019	#86-3020	#88-3054	#88-3059	#88-3061	#89-3117	#89-3117	#89-3124	#89-3124	#89-3124
	#89-3124	#89-3132	#89-3134	#89-3135	#89-3136	#89-3137	#89-3152	#89-3153	#89-3163	#89-3176
	#91-3202	#91-3202	#91-3202	#91-3202	#91-3207	#91-3227	#91-3228	#91-3234	#91-3247	#91-3248
	#91-3285	#91-3315	#93-3332	#93-3332	#93-3332	#93-3332	#93-3334	#93-3345	#93-3346	#93-3354
	#93-3357	#95-3369	#95-3369	#95-3369	#95-3369	#95-3374	#95-3395	#97-3405	#97-3405	#97-3405
	#97-3405	#97-3413	#97-3433	#99-3455	#99-3455	#99-3455	#99-3455	#99-3478	#102-3540	#102-3540
	#102-3543	#102-3543	#102-3543	#102-3543	#102-3543	#102-3543	#102-3545	#102-3552	#104-3557	#104-3557
	#104-3557	#104-3557	#104-3557	#104-3557	#104-3559	#104-3579	#106-3587	#106-3587	#106-3587	#106-3587
	#106-3587	#106-3587	#106-3589	#106-3621	#108-3627	#108-3627	#108-3627	#108-3627	#108-3627	#108-3627
	#108-3629	#108-3653	#110-3658	#110-3658	#110-3658	#110-3658	#110-3658	#110-3658	#110-3660	#110-3684
	#112-3689	#112-3689	#112-3689	#112-3689	#112-3689	#112-3689	#112-3691	#112-3715	#114-3720	#114-3720
	#114-3720	#114-3720	#114-3720	#114-3720	#114-3722	#114-3746	#116-3751	#116-3751	#116-3751	#116-3751
	#116-3751	#116-3751	#116-3758	#116-3848	#116-3855	#116-3857	#116-3859	#116-3868	#122-3984	#122-3984
	#122-3995	#122-3995	#122-3995	#122-3995	#124-4038	#124-4038	#124-4038	#124-4038	#126-4075	#126-4075
	#126-4076	#126-4076	#126-4076	#126-4076						
MSLDRO	#72-2714	72-2714	#82-2894	82-2894	#82-2935	82-2935	#82-2937	82-2937	#91-3207	91-3207
	#91-3234	91-3234	#93-3345	93-3345	#93-3357	93-3357				
MSMCHI	#5-451	#5-451								
MSMCLO	#5-451	#5-451								
MSPOP	#6-507	6-507	#8-552	8-552	#9-578	9-578	#9-580	9-580	#74-2744	74-2744
	#84-2948	84-2948	#84-2961	84-2961	#88-3073	88-3073	#89-3176	89-3176	#89-3192	89-3192
	#91-3315	91-3315	#93-3346	93-3346	#95-3395	95-3395	#97-3433	97-3433	#99-3478	99-3478
	#102-3552	102-3552	#104-3579	104-3579	#106-3621	106-3621	#108-3653	108-3653	#110-3684	110-3684
	#112-3715	112-3715	#114-3746	114-3746	#116-3868	116-3868	#120-3919	120-3919	#122-4010	122-4010
	#124-4052	124-4052	#126-4073	126-4073						
MSPRIN	#68-2480	#68-2480	#68-2485	#68-2485	#68-2488	#68-2488	#68-2490	#68-2490	#68-2504	#68-2504
	#68-2511	#68-2511	#68-2519	#68-2519	#74-2734	#74-2734	#74-2737	#74-2737	#74-2740	#74-2740
	#86-3020	#86-3020	#88-3054	#88-3054	#88-3059	#88-3059	#88-3061	#88-3061	#89-3134	#89-3134
	#89-3136	#89-3136	#89-3152	#89-3152	#89-3153	#89-3153	#89-3163	#89-3163	#93-3354	#93-3354
	#97-3413	#97-3413	#116-3848	#116-3848	#116-3855	#116-3855	#116-3857	#116-3857	#116-3859	#116-3859
MSPUSH	#5-477	#5-477	#6-503	#6-503	#8-539	#8-539	#9-561	#9-561	#10-633	#10-633
	#74-2727	#74-2727	#84-2944	#84-2944	#84-2952	#84-2952	#89-3117	#89-3117	#89-3124	#89-3124
	#91-3202	#91-3202	#93-3332	#93-3332	#95-3369	#95-3369	#97-3405	#97-3405	#99-3455	#99-3455
	#102-3540	#102-3540	#102-3543	#102-3543	#104-3557	#104-3557	#106-3587	#106-3587	#108-3627	#108-3627
	#110-3658	#110-3658	#112-3689	#112-3689	#114-3720	#114-3720	#116-3751	#116-3751	#122-3984	#122-3984
	#122-3995	#122-3995	#124-4038	#124-4038						
MSPUT	#68-2480	#68-2480	#68-2480	#68-2480	#68-2485	#68-2485	#68-2485	#68-2485	#68-2488	#68-2488
	#68-2488	#68-2490	#68-2490	#68-2490	#68-2504	#68-2504	#68-2504	#68-2511	#68-2511	#68-2511
	#68-2511	#68-2519	#68-2519	#68-2519	#74-2734	#74-2734	#74-2734	#74-2734	#74-2734	#74-2734

MACRO NAME	REFERENCES									
	#74-2737	#74-2737	#74-2737	#74-2737	#74-2737	#74-2737	#74-2737	#74-2740	#74-2740	#74-2740
	#74-2740	#82-2895	#82-2895	#82-2895	#82-2895	#82-2895	#82-2897	#82-2897	#82-2897	#82-2897
	#82-2897	#86-3020	#86-3020	#86-3020	#86-3020	#88-3054	#88-3054	#88-3054	#88-3054	#88-3059
	#88-3059	#88-3059	#88-3061	#88-3061	#88-3061	#89-3134	#89-3134	#89-3134	#89-3136	#89-3136
	#89-3136	#89-3152	#89-3152	#89-3152	#89-3152	#89-3153	#89-3153	#89-3153	#89-3153	#89-3153
	#89-3153	#89-3153	#89-3153	#89-3153	#89-3153	#89-3153	#89-3163	#89-3163	#89-3163	#89-3163
	#89-3163	#89-3163	#89-3163	#89-3163	#89-3163	#89-3163	#89-3163	#93-3334	#93-3334	#93-3334
	#93-3334	#93-3334	#93-3354	#93-3354	#93-3354	#97-3413	#97-3413	#97-3413	#97-3413	#116-3848
	#116-3848	#116-3848	#116-3855	#116-3855	#116-3855	#116-3855	#116-3857	#116-3857	#116-3857	#116-3857
	#116-3859	#116-3859	#116-3859	#116-3859						
M$PUT1	#68-2480	#68-2480	#68-2480	#68-2480	#68-2480	#68-2480	#68-2485	#68-2485	#68-2485	#68-2485
	#68-2485	#68-2485	#68-2488	#68-2488	#68-2488	#68-2488	#68-2490	#68-2490	#68-2490	#68-2490
	#68-2504	#68-2504	#68-2504	#68-2504	#68-2511	#68-2511	#68-2511	#68-2511	#68-2511	#68-2511
	#68-2519	#68-2519	#68-2519	#68-2519	#74-2734	#74-2734	#74-2734	#74-2734	#74-2734	#74-2734
	#74-2734	#74-2734	#74-2734	#74-2734	#74-2737	#74-2737	#74-2737	#74-2737	#74-2737	#74-2737
	#74-2737	#74-2737	#74-2737	#74-2737	#74-2737	#74-2737	#74-2740	#74-2740	#74-2740	#74-2740
	#74-2740	#74-2740	#82-2895	#82-2895	#82-2895	#82-2895	#82-2895	#82-2895	#82-2895	#82-2895
	#82-2897	#82-2897	#82-2897	#82-2897	#82-2897	#82-2897	#82-2897	#82-2897	#86-3020	#86-3020
	#86-3020	#86-3020	#86-3020	#86-3020	#88-3054	#88-3054	#88-3054	#88-3054	#88-3054	#88-3054
	#88-3059	#88-3059	#88-3059	#88-3059	#88-3061	#88-3061	#88-3061	#88-3061	#89-3134	#89-3134
	#89-3134	#89-3134	#89-3136	#89-3136	#89-3136	#89-3136	#89-3152	#89-3152	#89-3152	#89-3152
	#89-3152	#89-3152	#89-3153	#89-3153	#89-3153	#89-3153	#89-3153	#89-3153	#89-3153	#89-3153
	#89-3153	#89-3153	#89-3153	#89-3153	#89-3153	#89-3153	#89-3153	#89-3153	#89-3153	#89-3153
	#89-3153	#89-3153	#89-3163	#89-3163	#89-3163	#89-3163	#89-3163	#89-3163	#89-3163	#89-3163
	#89-3163	#89-3163	#89-3163	#89-3163	#89-3163	#89-3163	#89-3163	#89-3163	#89-3163	#89-3163
	#89-3163	#89-3163	#93-3334	#93-3334	#93-3334	#93-3334	#93-3334	#93-3334	#93-3334	#93-3334
	#93-3354	#93-3354	#93-3354	#93-3354	#97-3413	#97-3413	#97-3413	#97-3413	#97-3413	#97-3413
	#116-3848	#116-3848	#116-3848	#116-3848	#116-3855	#116-3855	#116-3855	#116-3855	#116-3855	#116-3855
	#116-3857	#116-3857	#116-3857	#116-3857	#116-3857	#116-3857	#116-3859	#116-3859	#116-3859	#116-3859
	#116-3859	#116-3859								
M$RADI	#122-3998	#122-3998	#122-3999	#122-3999	#122-4000	#122-4000	#122-4001	#122-4001	#122-4002	#122-4002
	#124-4039	#124-4039	#124-4040	#124-4040	#124-4041	#124-4041	#124-4042	#124-4042	#124-4043	#124-4043
	#124-4044	#124-4044	#124-4045	#124-4045						
M$RNRO	#91-3234	#91-3234	#91-3285	#91-3285						
M$SETS	#5-477	#5-477	#6-503	#6-503	#8-539	#8-539	#9-561	#9-561	#10-633	#10-633
	#74-2727	#74-2727	#84-2944	#84-2944	#84-2952	#84-2952	#89-3117	#89-3117	#89-3124	#89-3124
	#91-3202	#91-3202	#93-3332	#93-3332	#95-3369	#95-3369	#97-3405	#97-3405	#99-3455	#99-3455
	#102-3540	#102-3540	#102-3543	#102-3543	#104-3557	#104-3557	#106-3587	#106-3587	#108-3627	#108-3627
	#110-3658	#110-3658	#112-3689	#112-3689	#114-3720	#114-3720	#116-3751	#116-3751	#122-3984	#122-3984
	#122-3995	#122-3995	#124-4038	#124-4038						
M$SVC	#47-1736	47-1736	49-1823	#49-1824	49-1824	#49-1826	49-1826	#52-1947	52-1947	#62-2252
	62-2252	#68-2480	68-2480	#68-2485	68-2485	#68-2488	68-2488	#68-2490	68-2490	#68-2504
	68-2504	#68-2511	68-2511	#68-2519	68-2519	72-2668	72-2682	72-2688	72-2695	72-2706
	72-2710	#72-2714	72-2714	#74-2734	74-2734	#74-2737	74-2737	#74-2740	74-2740	#74-2744
	74-2744	#82-2879	82-2879	#82-2894	82-2894	#82-2895	82-2895	#82-2897	82-2897	#82-2935
	82-2935	#82-2937	82-2937	#84-2969	84-2969	#84-2970	84-2970	86-3019	#86-3020	86-3020
	#88-3054	88-3054	#88-3059	88-3059	#88-3061	88-3061	#89-3132	89-3132	#89-3134	89-3134
	#89-3135	89-3135	#89-3136	89-3136	#89-3137	89-3137	#89-3152	89-3152	#89-3153	89-3153
	#89-3163	89-3163	#89-3176	89-3176	#91-3207	91-3207	91-3227	#91-3228	91-3228	#91-3234
	91-3234	91-3247	#91-3248	91-3248	#91-3285	91-3285	#91-3315	91-3315	#93-3334	93-3334
	#93-3345	93-3345	#93-3346	93-3346	#93-3354	93-3354	#93-3357	93-3357	#95-3374	95-3374
	#95-3395	95-3395	#97-3413	97-3413	#97-3433	97-3433	#99-3478	99-3478	#102-3545	102-3545

MACRO NAME	REFERENCES									
	#102-3552	102-3552	#104-3559	104-3559	#104-3579	104-3579	#106-3589	106-3589	#106-3621	106-3621
	#108-3629	108-3629	#108-3653	108-3653	#110-3660	110-3660	#110-3684	110-3684	#112-3691	112-3691
	#112-3715	112-3715	#114-3722	114-3722	#114-3746	114-3746	#116-3758	116-3758	#116-3848	116-3848
	#116-3855	116-3855	#116-3857	116-3857	#116-3859	116-3859	#116-3868	116-3868		
MSTLAB	#47-1736	#49-1823	#49-1824	#49-1826	#52-1947	#62-2252	#68-2480	#68-2485	#68-2488	#68-2490
	#68-2504	#68-2511	#68-2519	#72-2668	#72-2682	#72-2688	#72-2695	#72-2706	#72-2710	#72-2714
	#74-2734	#74-2737	#74-2740	#74-2744	#82-2879	#82-2894	#82-2895	#82-2897	#82-2935	#82-2937
	#84-2969	#84-2970	#86-3019	#86-3020	#88-3054	#88-3059	#88-3061	#89-3132	#89-3134	#89-3135
	#89-3136	#89-3137	#89-3152	#89-3153	#89-3163	#89-3176	#91-3207	#91-3227	#91-3228	#91-3234
	#91-3247	#91-3248	#91-3285	#91-3315	#93-3334	#93-3345	#93-3346	#93-3354	#93-3357	#95-3374
	#95-3395	#97-3413	#97-3433	#99-3478	#102-3545	#102-3552	#104-3559	#104-3579	#106-3589	#106-3621
	#108-3629	#108-3653	#110-3660	#110-3684	#112-3691	#112-3715	#114-3722	#114-3746	#116-3758	#116-3848
	#116-3855	#116-3857	#116-3859	#116-3868						
MSTSTL	#47-1736	47-1736	#49-1823	#49-1823	49-1823	#49-1824	49-1824	#49-1826	49-1826	#52-1947
	52-1947	#62-2252	62-2252	#68-2480	68-2480	#68-2485	68-2485	#68-2488	68-2488	#68-2490
	68-2490	#68-2504	68-2504	#68-2511	68-2511	#68-2519	68-2519	#72-2668	#72-2668	72-2668
	#72-2682	#72-2682	72-2682	#72-2688	#72-2688	72-2688	#72-2695	#72-2695	72-2695	#72-2706
	#72-2706	72-2706	#72-2710	#72-2710	72-2710	#72-2714	72-2714	#74-2734	74-2734	#74-2737
	74-2737	#74-2740	74-2740	#74-2744	74-2744	#82-2879	82-2879	#82-2894	82-2894	#82-2895
	82-2895	#82-2897	82-2897	#82-2935	82-2935	#82-2937	82-2937	#84-2969	84-2969	#84-2970
	84-2970	#86-3019	#86-3019	86-3019	#86-3020	86-3020	#88-3054	88-3054	#88-3059	88-3059
	#88-3061	88-3061	#89-3132	89-3132	#89-3134	89-3134	#89-3135	89-3135	#89-3136	89-3136
	#89-3137	89-3137	#89-3152	89-3152	#89-3153	89-3153	#89-3163	89-3163	#89-3176	89-3176
	#91-3207	91-3207	#91-3227	#91-3227	91-3227	#91-3228	91-3228	#91-3234	91-3234	#91-3247
	#91-3247	91-3247	#91-3248	91-3248	#91-3285	91-3285	#91-3315	91-3315	#93-3334	93-3334
	#93-3345	93-3345	#93-3346	93-3346	#93-3354	93-3354	#93-3357	93-3357	#95-3374	95-3374
	#95-3395	95-3395	#97-3413	97-3413	#97-3433	97-3433	#99-3478	99-3478	#102-3545	102-3545
	#102-3552	102-3552	#104-3559	104-3559	#104-3579	104-3579	#106-3589	106-3589	#106-3621	106-3621
	#108-3629	108-3629	#108-3653	108-3653	#110-3660	110-3660	#110-3684	110-3684	#112-3691	112-3691
	#112-3715	112-3715	#114-3722	114-3722	#114-3746	114-3746	#116-3758	116-3758	#116-3848	116-3848
	#116-3855	116-3855	#116-3857	116-3857	#116-3859	116-3859	#116-3868	116-3868		
MSWORD	#5-494	#5-494	#7-522	#7-522	#7-522	#7-522	#7-522	#7-522	#7-522	#7-522
	#7-522	#7-522	#49-1823	#49-1823	#49-1823	#49-1823	#72-2668	#72-2668	#72-2668	#72-2668
	#72-2682	#72-2682	#72-2682	#72-2682	#72-2688	#72-2688	#72-2688	#72-2688	#72-2695	#72-2695
	#72-2695	#72-2695	#72-2706	#72-2706	#72-2706	#72-2706	#72-2710	#72-2710	#72-2710	#72-2710
	#86-3019	#86-3019	#86-3019	#86-3019	#91-3227	#91-3227	#91-3227	#91-3227	#91-3247	#91-3247
	#91-3247	#91-3247	#102-3545	#104-3559	#106-3589	#108-3629	#110-3660	#112-3691	#114-3722	#116-3758
	#122-3998	#122-3998	#122-3999	#122-3999	#122-4000	#122-4000	#122-4001	#122-4001	#122-4002	#122-4002
	#124-4039	#124-4039	#124-4040	#124-4040	#124-4041	#124-4041	#124-4042	#124-4042	#124-4043	#124-4043
	#124-4044	#124-4044	#124-4045	#124-4045	#126-4076	#126-4076				
POINTE	5-484									
POP	#23-1070	43-1695	43-1696	52-1948	52-1956	58-2138	60-2189	60-2190	68-2531	68-2532
	72-2715	72-2716	72-2717	72-2718	74-2742	74-2743	78-2808	78-2809	80-2846	80-2847
	86-3024	86-3025	86-3026	88-3062	88-3063	89-3170	89-3171	89-3172	89-3173	89-3174
	89-3175	97-3411	97-3412							
PRINTB	#68-2504	#74-2734	#74-2737	#74-2740	#86-3020					
PRINTF	88-3054	88-3059	88-3061	93-3354	97-3413	116-3848	116-3855	116-3857	116-3859	
PRINTS	#89-3134	#89-3136	#89-3152	#89-3153	#89-3163					
PRINTX	68-2480	68-2485	68-2488	68-2490	68-2511	68-2519				
PUSH	#23-1057	43-1688	43-1689	52-1942	52-1946	58-2128	60-2178	60-2179	68-2456	68-2457
	72-2652	72-2653	72-2654	72-2655	74-2728	74-2729	78-2784	78-2785	80-2824	80-2825
	86-2998	86-2999	86-3000	88-3047	88-3048	89-3125	89-3126	89-3127	89-3128	89-3129

MACRO NAME	REFERENCES							
	89-3130	97-3407	97-3408					
READEF	#91-3207							
RFLAGS	#91-3285							
SETPRI	82-2894							
SETVEC	82-2895	82-2897	93-3334					
SVC	#5-450	5-451						
SWAPIN	#23-1087	#49-1773	#49-1784	#49-1791	#49-1812			
SWAPOW	#23-1108	50-1869						
TSTID	#35-1512	102-3544	104-3558	106-3588	108-3628	110-3659	112-3690	114-3721
TUREAD	#31-1394	#106-3597	#110-3670	#114-3732				
TURTRY	#29-1319	#106-3596	#106-3597	#108-3639	#110-3670	#112-3701	#114-3732	
TUSEEK	#27-1257	104-3565						
TUSELF	#33-1471	#102-3547						
TUWRIT	#25-1149	106-3596	108-3639	112-3701				