TU58

TU58 PERF EXERCISER

CZTUUB0

AH E649B M(

FICHE 1 OF 1

NOV 1979

MADE IN USA

.REM &

IDENTIFICATION

PRODUCT CODE: AC-E648B-MC

PRODUCT NAME: CZTUUBO TU58 PERF EXER

PRODUCT DATE: JULY 1979

MAINTAINER: DIAGNOSTIC ENGINEERING GROUP

AUTHOR: R. J. ROSS

TABLE OF CONTENTS

1.0 GENERAL INFORMATION
1.1 PROGRAM ABSTRACT
1.2 SYSTEM REQUIREMENTS
1.3 RELATED DOCUMENTS AND STANDARDS
1.4 DIAGNOSTIC HIERARCHY PREREQUISITES
1.5 ASSUMPTIONS

2.0 OPERATING INSTRUCTIONS
2.1 HOW TO RUN THIS DIAGNOSTIC

3.0 ERROR INFORMATION

4.0 PERFORMANCE AND PROGRESS REPORTS

5.0 DEVICE INFORMATION TABLES

6.0 TEST SUMMARIES

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 IMPLIMENTS 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 PROTOCOL 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 IMPLIMENT THE COMMUNICATIONS PROTOCOL SIMPLIFIES CONTEXT SWITCHING FROM UNIT TO UNIT BY NOT REQUIRING 8 SEPARATE DEVICE STACKS IN ADDITION TO THE SYSTEM STACK.
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 SCHEDULING 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. "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" SUCCES 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 INTERFACE; AND REVISION "I" TU58 MICROCODE (OR LATER) ASSUMED.

1.2.2 SOFTWARE

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

EST DRO - 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 A 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.

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

```
                                .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,B,0,3600.,1

002122                          DESCRIP <TU58 PERF EXER>
```

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

002142                          BGNPROT
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 7
```

```
                                .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.
                                ;--

002170                                  BGNHW   DFPTBL

002172  176500                         .WORD   176500          ;CSR ADDRESS
002174  000300                         .WORD   300             ;VECTOR ADDR.
002176  000003                         .WORD   3               ;TEST DRIVE ZERO AND ONE
002200  000000                         .WORD   0               ;NOT PDT TYPE INTERFACE

002202                                  ENDHW
```

```
                                        .SBTTL  SOFTWARE P-TABLE

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

        002202                                  BGNSW   SFPTBL

        002204  00 010                  LENGTH: .WORD   8.              ;TAPE LENGTH
        002206  000001                  STAEOP: .WORD   1               ;PRINT STATISTICS AT EOP
        002210  000001                  PRBUF:  .WORD   1               ;PRINT DATA BUF ON COMP. ERROR
        002212  000001                  CMPDAT: .WORD   1               ;COMPARE DATA
        002214  000001                  DRVCHK: .WORD   1               ;ADD DR # TO DATA
        002216  000001                  EVLTHR: .WORD   1               ;THRESHOLD FOR EVL TEST


        002220                                  ENDSW

        002220                                  ENDMOD
```

```
                                .TITLE GLOBAL AREAS
                                .SBTTL  GLOBAL EQUATES SECTION
                                
                                
002220                                  BGNMOD
                                
                                ;++
                                ; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
                                ; ARE USED IN MORE THAN ONE TEST.
                                ;--
                                
002220                                  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                          PR7== 34
000300                          PR6== 300
```

```
000240                  PRIO5== 240
000200                  PRIO4== 200
000140                  PRIO3== 140
000100                  PRIO2== 100
000040                  PRIO1== 40
000000                  PRIO0== 0
                        ;
                        ;OPERATOR FLAG BITS
                        ;
000004                  EVL==     4
000010                  LOT==    10
000020                  ALR==    20
000040                  IOJ==    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
```

```
                        .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
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.
000052                  OTL     ==      42.
000054                  NOART   ==      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

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
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
                        ;-------------------------------------------------------------------
```

```
                                .SBTTL  ERROR MESSAGE DESCRIPTIONS

                                ;THE TABLE OF ERROR MESSAGES (ADDRESSES).  ABNDX(R5) CONTAINS THE OFFSET
                                ;OF THE REASON.  IT'S ABSOLUTE ADDRESS IS RSNTAB + ABNDX(R5).

002220  002314                  RSNTAB: MSNLOG
002222  003046                          MSSFRD
002224  003106                          MSSFWR
002226  002530                          MSRNIT
002230  002314                          MSNLOG
002232  003252                          MSOVRN
002234  002376                          MSCOM
002236  003146                          MSHDRD
002240  003210                          MSHDWR
002242  002550                          MSHCHK
002244  002300                          MSSKER
002246  002506                          MSWPRO
002250  002440                          MSNOMO
002252  002612                          MSNIT
002254  002626                          MSPART
002256  002650                          MSUNIT
002260  002676                          MSCMD
002262  002712                          MSREC
002264  002356                          MSSELF
002266  002732                          MSWRSP
002270  002756                          MSNRSP
002272  002772                          MSQRSP
002274  002456                          MSNOTP
002276  003024                          MSTOSN
```

```
                                ;HERE ARE THE MESSAGES PROPER:

002300  123  105  105  MSSKER:: .ASCIZ /SEEK ERROR/            ;DEVICE COULD NOT READ HEADER
                              .EVEN
002314  123  131  123  MSNLOG:: .ASCIZ /SYSTEM ERROR/          ;DIAGNOSTIC HUNG. BETTER RE-BOOT
                              .EVEN
002332  102  101  104  MSBDA:: .ASCIZ  /BAD DATA IN PACKET/    ;HOST DATA CHECK FOUND ERROR, DEVICE MAY
                              .EVEN                            ;HAVE READ CORRECTLY.
002356  123  105  114  MSSELF:: .ASCIZ /SELF TEST ERROR/       ;MICRO DIAGNOSTIC FAILED, BUT DEVICE COULD STILL
                              .EVEN                            ;SEND AN END PACKET.
002376  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
002440  115  117  124  MSNOMO:: .ASCIZ /MOTOR STOPPED/         ;DEVICE COULD NOT GET ANY MEANINGFUL SIGNAL
                              .EVEN                            ;FROM TAPE.
002456  103  101  122  MSNOTP:: .ASCIZ /CARTRIDGE NOT IN PLACE/ ;NO MEDI - OR BAD SWITCH
                              .EVEN
002506  127  122  111  MSWPRO:: .ASCIZ /WRITE PROTECTION/       ;CARTRIDGE WRITE PROTECT TAB MISSING OR
                              .EVEN                            ;SWITCH BAD
002530  122  105  103  MSRNIT:: .ASCIZ /RECIEVING INIT/         ;DEVICE SENT INIT REQUEST
                              .EVEN
002550  110  117  123  MSHCHK:: .ASCIZ /HOST FOUND PACKET CHECKSUM ERROR/  ;DEVICE SENT PACK WITH
                              .EVEN                            ;BAD CHECKSUM
002612  103  101  116  MSNIT:: .ASCIZ  /CAN'T INIT/            ;DEVICE SENT BYTE OTHER THAN "CONTINUE"
                              .EVEN                            ;DURING INITIALIZATION
002626  120  101  122  MSPART:: .ASCIZ /PARTIAL OPERATION/     ;END OF MEDIUM ENCOUNTERED
                              .EVEN
002650  042  116  117  MSUNIT:: .ASCIZ /"NON-EXISTENT" DRIVE/  ;DEVICE RECV'D TOO LARGE DRIVE NUMBER
                              .EVEN
002676  102  101  104  MSCMD:: .ASCIZ  /BAD COMMAND/           ;DEVICE COULD NOT UNDERSTAND HOST
                              .EVEN
002712  102  101  104  MSREC::  ASCIZ  /BAD RECORD NO./        ;DEVICE RECV'D TOO LARGE A RECORD NUMBER
                              .EVEN
002732  127  122  117  MSWRSP:: .ASCIZ /WRONG SUCCESS CODE/    ;HOST COULD NOT DECIPHER CODE IN END PACK
                              .EVEN
002756  116  117  040  MSNRSP:: .ASCIZ /NO RESPONSE/           ;TIME OUT WAITING FOR BYTE IN RCV BUF ON INTERFACE.
                              .EVEN
002772  111  116  104  MSQRSP:: .ASCIZ \INDECIPHERABLE FLAG BYTE\  ;HOST COULD NOT UNDERSTAND 1ST BYTE OF
                              .EVEN                            ;RESPONSE FROM TU AS PROPER PROTOCOL
003024  124  111  115  MSTOSN:: .ASCIZ /TIME OUT ON SEND/      ;DLV 'READY' NEVER WENT HIGH
                              .EVEN
003046  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
003106  122  105  103  MSSFWR:: .ASCIZ /RECOV. DATA CHECK ERR ON WR OP/  ;SAME BUT WR OR WR VERIFY OPERATION
                              .EVEN
003146  125  116  122  MSHDRD:: .ASCIZ /UNRECOV. DATA CHECK ERR ON RD OP/  ;TU58 RESPONDED WITH "DATA-CHECK"
                              .EVEN                            ;ERROR ON READ OP.  ;RETRIES UNSUCCESSFUL
003210  125  116  122  MSHDWR:: .ASCIZ /UNRECOV. DATA CHECK ERR ON WR OP/  ;SAME BUT WR OPERATION
                              .EVEN
003252  104  114  126  MSOVRN:: .ASCIZ /DLV ERROR IN RECEIVE/  ;DLV ERROR (THE CONTENTS PRINTED OUT)
                              .EVEN
```

```
                                .SBTTL  MISC STORAGE AND EQUATES

003300  000000          SYSTAT:: .WORD                  ;SYSTEM STATUS WORD
                                                        ;BIT7-BIT15 = 1ST BYTE OF PACK RCVD
                                                        ;BIT05 = NOT USED
                                                        ;BIT04 = NOT USED
                                                        ;BIT03 = NOT USED
                                                        ;BIT02 = NOT USED
                                                        ;BIT01 = RETRY OCCURRING
                                                        ;BIT00 = CHKSUM IS ODD (TEMP STORAGE)

003302  000000          TAPLEN:: .WORD                  ;# RECORDS
003304  000000          DEVPTR:: .WORD                  ;->NEXT UNIT ADDRESS
003306  000000          RCFLG::  .WORD                  ;TEMP STORE FOR GTBYTE
003310  000000          RCBCNT:  .WORD                  ;TEMP STORAGE FOR GTBYTE
003312  000004          FTLNM:   .WORD 4                ;NUMBER OF TRIES BEFORE ABORT
003314  000000          DONE::   .WORD                  ;1 IF TEST ALGORITHM COMPLETE
003316  000000          IDPTR::  .WORD                  ;PTR IN TESTID MACRO
003320  000000          TSTTOP:  .WORD                  ;POINTER TO CURRENT TEST
003322  000010          MXRTRY:: .WORD  8.              ;NUMBER OF RETRIES, BEFORE "HARD" ERROR
003324  000000          BLKER::  .WORD                  ;RECORD # FOR ERROR REPORT
003326  000000          SECREC:: .WORD                  ;RECORD # TO START AFTER SWITCHING TRACKS
003330  000000          PRNSIZ:: .WORD                  ;SIZE OF PACK FOR "PRNPAK"
003332  000000          ALLGON:: .WORD                  ;SECONDARY DON'T PRINT STATISTICS FLAG.


                        ;               TIME OUT CONSTANTS:

003334  177777                        CSNRDY::  -1            ;TIME OUT ON SEND DELAY (DLV NOT READY)
003336  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 = HALTED
                                                        ;BIT12 = TEMP STOR WRITE MACRO
                                                        ;BIT11 = UNIT NOT BEING TESTED
                                                        ;BIT10 = RETRYING
                                                        ;BIT9  = TU58 CHKSUM ERROR
                                                        ;BIT8  = RD/WR OPERATION
                                                        ;BIT7  = NORMAL/REDUCED THRESHOLD (MACROS)
                                                        ;BIT6  = HOST DATA COMPARE ERROR
                                                        ;BIT5  = WR  ERIFY OPERATION
                                                        ;BIT4  = TYPE OF PAK SENT 0DATA 1CMD
                                                        ;BIT3  =  NOT USED
                                                        ;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                    ;**RESERVED**

                        ;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                    ;WEIRD FLAG     ; OTL   ;MSQRSP ;******
                        ;       WORD                    ;NO CARTRIDGE   ;NOCART ;MSNOTP ;ESNCRT
                        ;       WORD                    ;TIME OUT  SEND;TOSNDB  ;MSTOSN ;******

000202                  BLKEND  ==      130.            ;OFFSET OF END OF STATISTICS (RESERVED)
                        ;       WORD                    ;** RESERVED **
000204                  IUVECT  ==      132.            ;VECTOR ADDRESS
                        ;       WORD                    ;** RESERVED **
000210                  BLKSIZ  ==      136.            ;** RESERVED **
                        ;-------------------------------------------------------------
```

```
    900                             .SBTTL  DEVICE DATA BLOCK ALLOCATION
    901
    902
    903                             ;TABLE OF DEVICE DATA BLOCK ADDRESSES
    904
    905
    906 003340  003360              BLKTBL::        .WORD   DEV0
    907 003342  003570                              .WORD   DEV1
    908 003344  004000                              .WORD   DEV2
    909 003346  004210                              .WORD   DEV3
    910 003350  004420                              .WORD   DEV4
    911 003352  004630                              .WORD   DEV5
    912 003354  005040                              .WORD   DEV6
    913 003356  005250              LSTDEV::        .WORD   DEV7
    914
    915
    916                             ;AND STORAGE FOR EACH:
    917
    918 003360                      DEV0:           .BLKB   BLKSIZ
    919 003570                      DEV1:           .BLKB   BLKSIZ
    920 004000                      DEV2:           .BLKB   BLKSIZ
    921 004210                      DEV3:           .BLKB   BLKSIZ
    922 004420                      DEV4:           .BLKB   BLKSIZ
    923 004630                      DEV5:           .BLKB   BLKSIZ
    924 005040                      DEV6:           .BLKB   BLKSIZ
    925 005250                      DEV7:           .BLKB   BLKSIZ
```

```
    941                             .SBTTL  GLOBAL TEXT SECTION
    942
    943                             ;
    944                             ; NAMES OF DEVICES SUPPORTED BY PROGRAM
    945                             ;
    946 00546C                              DEVTYP  <TU58 CONTROLLER>
    947
    959
    960
    978
```

```
                        .SBTTL  SYSTEM MACRO DEFINITIONS

                                .MACRO  PUSH    ,REG
                                        MOV     REG,-(SP)
                                .ENDM

                                .MACRO  POP,REG
                                        MOV     (SP)+,REG
                                .ENDM

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

                                .MACRO  SWAPIN
                                        MOV     6.(R5),R0
                                        MOV     8.(R5),R1
                                        MOV     10.(R5),R2
                                        MOV     12.(R5),R3
                                        MOV     14.(R5),R4
                                .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
                                        MOV     R0,6.(R5)
                                        MOV     R1,8.(R5)
                                        MOV     R2,10.(R5)
                                        MOV     R3,12.(R5)
                                        MOV     R4,14.(R5)
                                .ENDM
```

```
                        ;++
                        ;THE WRITE MACRO IMPLEMENTS THE COMPLETE PROTOCOL NECESSARY TO BUILD
                        ;A COMMAND PACKET AND SUBSEQUENT DATA PACKETS (UNTIL THE BYTE COUNT
                        ;(BCNT) IS SATISFIED).
                        ;
                        ;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
                        ;           UNIT'S 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  TUWRIT  PTRN,REC,BCNT,DR,VER,?A,?B,?C,?D,?E,?F,?G,?H,?I
                        :       MOV     #TRBUF,R0       ;MAKE COMMAND PACKET:
                                MOVB    #RSCMND,@R0     ;COMMAND FLAG
                                MOVB    #RSMSIZ,1(R0)   ;THIS SIZE
                                MOVB    #RSSWR,2(R0)    ;INSERT OP CODE-WRITE
                                MOVB    VER,3.(R0)      ;VERIFY (1 OR 0)
                                MOVB    DR,4.(R0)       ;DRIVE #
                                MOVB    #020,5.(R0)     ;MAINTENANCE MODE SWITCH
                                CLR     6.(R0)          ;NO SEQUENCE #
                                MOV     BCNT,8.(R0)     ;TOTAL COUNT TO WRITE
                                MOV     REC,10.(R0)     ;AT  RECORD N
                                MOV     #RSMSIZ,R1      ;THE PACKET SIZE PLUS+2
                                TST     (R1)+           ;(FLAG AND COUNT) INTO R1
                                MOV     #RSSNSZ,SNDCNT(R5)  ;LOAD THE  SIZE TO SEND
                                CALL    CHKSUM          ;R0 --> R1=COUNT
                                MOV     R1,(R0)         ;PUT CHKSUM IN PACKET
                                                        ;SET UP EXPECTATIONS:
                                MOV     #RSCONT,XSFLG(R5)   ;THE FLAG
                                MOV     #1,XSCNT(R5)    ;THE COUNT
                                MOV     #1,XSPKNM(R5)   ;THE # PACKETS EXPECTED
                                CALL    RSVP            ;SEND (AND RETURN TO SCHEDULER)
                                BIC     #BIT12,@R5      ;FLAG FOR LAST PACKET
                                MOV     BCNT,R2         ;GET # OF DATA BYTES
                A:              MOV     #TRBUF,R0       ;POINT TO TOP OF BUFFER AGAIN
                                CMP     R2,#128.        ;START DATA PACKET(S)
                                BHI     B               ;BCNT > 128.!
                                MOV     R2,R1           ;BCNT<128.
                                BIS     #BIT12,@R5      ;SO LAST PACKET NOW
                                BR      C               ;USE REMAINING COUNT
                B:              MOV     #128.,R1        ;USE 128. BYTES
                C:              MOVB    R1,1(R0)        ;COPY COUNT TO BUFFER
                                MOV     R1,R3           ;R3=COUNTER TO LOAD BUFFER
                                MOVB    #RSDATA,@R0     ;FLAG FIRST
```

```
1081                                        1ST     (R0)+           ;SKIP COUNT
1082                                D:      MOVB    PTRN,(R0)+      ;INSERT DATA
1083                                        DEC     R3              ;MORE?
1084                                        BHI     D               ;YES
1085                                        MOV     #TRBUF,R0       ;-->TOP AGAIN
1086                                        MOVB    1(R0),R1        ;GET COUNT
1087                                        BIC     #177400,R1      ;ZERO SIGN EXTEND
1088                                        MOV     R1,SNDCNT(R5)   ;HOW MANY TO SEND PLUS
1089                                        ADD     #4,SNDCNT(R5)   ;FLAG,COUNT,CHKSUM
1090                                        ADD     #2,R1           ;COMPENSATE FOR FLAG + COUNT
1091                                        CALL    CHKSUM          ;FOR CHECKSUM CALC.
1092                                        MOVB    R1,(R0)+        ;CHKSUM INTO PACKET
1093                                        SWAB    R1              ;EVEN ON AN ODD
1094                                        MOVB    R1,(R0)+        ;BYTE BOUNDARY
1095                                        BIT     #BIT12,@R5      ;LAST DATA PACKET?
1096                                        BEQ     E               ;NO
1097                                        MOV     #RSEND,XSFLG(R5)  ;YES-EXPECT 'END'
1098                                        MOV     #RSNDSZ,XSCNT(R5)  ;OF THIS SIZE
1099                                        MOV     #1,XSPKNM(R5)   ;AND 1 PACKET
1100                                        BR      F               ;SEND
1101                                E:      MOV     #RSCONT,XSFLG(R5)  ;(NOT LAST), EXPECT 'CONTINUE'
1102                                        MOV     #1,XSCNT(R5)    ;AND 1 BYTE
1103                                        MOV     #1,XSPKNM(R5)   ;AND 1 PACKET
1104                                F:      CALL    RSVP            ;SEND PACKET
1105                                                                ;AND RETURN TO SCHEDULER
1106                                        BIT     #BIT10,@R5      ;RETRY?
1107                                        BNE     G               ;YES
1108                                        SUB     #128.,R2        ;NO, MORE DATA TO SEND?
1109                                        BHI     A               ;YES
1110                                        BR      H               ;NO
1111                                G:      TURTRY  REC,BCNT,DR     ;RETRY HERE
1112                                        BIT     #BIT10,@R5      ;RETRY AGAIN?
1113                                        BNE     G               ;YES
1114                                H:      NOP                     ;DONE
1115                    .ENDM
```

```
                        ;++
                        ;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
                        
                                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 SEND PACK
                                MOV     #RSNDSZ,XSCNT(R5)  ;COUNT WITH THIS
                                MOV     #1.,XSPKNM(R5)  ;EXPECT ONLY 1 PACKET
                        
                                CALL    RSVP            ;SEND
                                                        ;AND RETURN TO SCHEDULER
                        .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

                        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
                                MOV     #132.,(R3)+     ;AND EXPECT COUNT
                                SUB     #128.,R1        ;NEG RESULT LAST TIME
                                BLOS    .               ;LAST TIME!
                                BR      A               ;MORE TO DO
```

```
1228                                                    C:      INC     R2              ;ADD ONE FOR END PACK
1229                                                            MOV     R2,xSPKNM(R5)   ;SAVE # PACKETS TO EXPECT
1230                                                            MOV     #RSEND,(R3)+    ;EXPECT AN END
1231                                                            MOV     #RSNDSZ,(R3)    ;THIS BIG-14. BYTES
1232
1233                                                            CALL    RSVP            ;SEND
1234                                                                                    ;AND RETURN TO SCHEDULER
1235                                            .ENDM
```

```
1238                            ;++
1239                            ;THE READ MACRO IMPLIMENTS THE COMPLETE PROTOCOL NECESSARY TO INITIATE
1240                            ;A READ SEQUENCE.
1241                            ;
1242                            ;SETS UP THE EXPECTED PROTOCOL RESPONSES:  THE NUMBER OF PACKETS
1243                            ;(XSPKNM) AND THEIR FLAG BYTES AND COUNTS (XSFLG, XSCNT).  CALLS
1244                            ;'RSVP' TO SEND EACH PACKET, AND 'CHKSUM' TO CALC. THE PACKET
1245                            ;CHECKSUM.
1246                            ;
1247                            ; INPUTS  - DEVICE BLOCK @R5
1248                            ;           TRBUF - BUFFER ADDRESS
1249                            ;           UNITS TEST REGISTERS FROM SWAPIN
1250                            ;
1251                            ; OUTPUTS - SNDCNT(R5) = # OF BYTES TO SEND
1252                            ;           XSPKNM = # OF PACKETS EXPECTED
1253                            ;           XSFLG = FLAG BYTE OF 1ST PACKET
1254                            ;           XSCNT = BYTE COUNT OF 1ST PACKET
1255                            ;             . ***
1256                            ;             .   *  SUBSEQUENT XSFLGS
1257                            ;             .    >
1258                            ;             .   *  AND XSCNTS
1259                            ;             . ***
1260                            ;--
1261
1262                                    .MACRO  TUREAD  REC,BCNT,DR,VER,?A,?B,?C,?D
1263
1264
1265                                            MOV     #TRBUF,R0       ;FORM CMND PACK:
1266                                            MOVB    #RSCMND,@R0     ;MESSAGE PACK TYPE
1267                                            MOVB    #RSMSIZ,1(R0)   ;THIS BIG
1268                                            MOVB    #RSSRD,2(R0)    ;OP CODE IS READ
1269                                            MOV     REC,10.(R0)     ;THIS RECORD
1270                                            MOVB    DR,4.(R0)       ;THIS DRIVE
1271                                            MOVB    VER,3.(R0)      ;VERIFY
1272                                            MOV     BCNT,8.(R0)     ;TOTAL BYTES TO READ
1273                                            MOVB    #020,5.(R0)     ;MAINTENANCE MODE
1274                                            CLR     6.(R0)          ;NO SEQUENCE #
1275                                            MOV     #RSMSIZ,R1      ;GET SIZE OF PACKET
1276                                            TST     (R1)+           ;+2 FOR CHECKSUM
1277                                            MOV     #RSSNSZ,SNDCNT(R5)  ;SIZE TO SEND
1278                                            CALL    CHKSUM          ;FORM CHECKSUM R1=COUNT
1279                                            MOV     R1,(R0)         ;INSERT CHECKSUM
1280
1281                                            MOV     BCNT,R1         ;SET EXPECTATIONS:
1282                                                                    ;CALC # OF DATA PACKETS TO EXPECT:
1283                                            MOV     #XSFLG,R3       ;GET OFFSET
1284                                            ADD     R5,R3           ;ABS. ADDR. OF XSFLG
1285                                            CLR     R2              ;PRESET AS NONE
1286                                    A:      INC     R2              ;# PACKETS EXPECTED
1287                                            MOV     #RSDATA,(R3)+   ;LOAD XSFLG
1288                                            MOV     #132.,(R3)+     ;AND EXPECTED COUNT
1289                                            SUB     #128.,R1        ;NEG RESULT LAST TIME
1290                                            BLOS    C               ;LAST TIME
1291                                            BR      A               ;MORE TO DO
1292                                    C:      INC     R2              ;ADD ONE FOR END PACK
1293                                            MOV     R2,XSPKNM(R5    ;SAVE # PACKETS TO EXPECT
1294                                            MOV     #RSEND,(R3)+    ;EXPECT AN END ALSO...
```

```
1295                                                    MOV     #RSNDSZ,(R3)    ;THIS BIG-14. BYTES
1296                                                    CALL    RSVP            ;SEND
1297                                                                            ;AND RETURN TO SCHEDULER
1298                                            D:      BIT     #BIT10,@R5      ;RETRY?
1299                                                    BEQ     B               ;NO.
1300                                                    TURTRY  REC,BCNT,DR     ;YES
1301                                                    BR      D               ;ANOTHER RETRY?
1302                                            B:      NOP                     ;NO
1303
1304                                    .ENDM
```

```
1307                            ;++
1308                            ;THE SELF TEST MACRO IMPLIMENTS THE COMPLETE PROTOCOL NECESSARY TO
1309                            ;INITIATE A 'DIAGNOSE' SEQUENCE.
1310                            ;
1311                            ;SETS UP THE EXPECTED PROTOCOL RESPONSES:  THE NUMBER OF PACKETS
1312                            ;(XSPKNM) AND THEIR FLAG BYTES AND COUNTS (XSFLG, XSCNT).  CALLS
1313                            ;'RSVP' TO SEND EACH PACKET, AND 'CHKSUM' TO CALC. THE PACKET
1314                            ;CHECKSUM.
1315                            ;
1316                            ; INPUTS  - DEVICE BLOCK @R5
1317                            ;           TRBUF - BUFFER ADDRESS
1318                            ;           UNITS REGISTERS TEST FROM SWAPIN
1319                            ;
1320                            ; OUTPUTS - SNDCNT(R5) = # OF BYTES TO SEND
1321                            ;           XSPKNM = # OF PACKETS EXPECTED
1322                            ;           XSFLG = FLAG BYTE OF 1ST PACKET
1323                            ;           XSCNT = BYTE COUNT OF 1ST PACKET
1324                            ;             . ***
1325                            ;             .   *  SUBSEQUENT XSFLGS
1326                            ;             .    >
1327                            ;             .   *  AND XSCNTS
1328                            ;             . ***
1329                            ;--
1330
1331                                    .MACRO  TUSELF
1332
1333                                                    MOV     #TRBUF,R0       ;FORM COMMAND PACKET
1334                                                    MOVB    #RSCMND,@R0     ;COMMAND FLAG
1335                                                    MOVB    #RSMSIZ,1(R0)   ;SIZE OF MESSAGE
1336                                                    MOVB    #RSSSLF,2(R0)   ;SELF TEST OPERATION
1337                                                    CLRB    3(R0)           ;NO MODIFIER.
1338                                                    CLR     4(R0)           ;NO DRIVE OR SWITCHES
1339                                                    CLR     6(R0)           ;NO SEQUENCE NUMBER
1340                                                    CLR     8.(R0)          ;NO BYTES
1341                                                    CLR     10.(R0)         ;NO RECORD #
1342                                                    MOV     #RSMSIZ,R1      ;GET SIZE
1343                                                    TST     (R1)+           ;+2 FOR CHECKSUM
1344                                                    MOV     #RSSNSZ,SNDCNT(R5)  ;SIZE TO SEND
1345                                                    CALL    CHKSUM          ;FORM CHECKSUM
1346
1347                                                    MOV     R1,(R0)         ;INSERT INTO PACKET
1348                                                    MOV     #RSEND,XSFLG(R5)  ;EXPECT END.
1349                                                    MOV     #RSNDSZ,XSCNT(R5)  ;THIS BIG
1350                                                    MOV     #1,XSPKNM(R5)   ;AND 1 PACKET
1351                                                                            ;SEND
1352                                                    CALL    RSVP            ;RETURN TO SCHEDULER
1353
1354                                    .ENDM
```

```
                        ;++
                        ;THE TEST ID MACRO INTERFACES THE SUPERVISOR'S TEST DISPATCH TO THE
                        ;DIAGNOSTIC'S FORMAT BY IMPLEMENTING CALLS THAT:  1) INITIALIZE THE
                        ;PC OF THE TEST CODE (TSTPC(R5)),  2) ASSIGN THE 1ST DRIVES,  3) RUN
                        ;THE TEST,  4) SWITCH DRIVES AND REINITIALIZE, 5) RUN THE TEST AGAIN.
                        ;--

                                .MACRO  TSTID   ADDRES,?A

                                .NLIST
                                .LIST ME
                                .LIST
                                                MOV     ADDRES,TSTTOP   ;SAVE ADDR OF TEST
                                                CALL    SETUP           ;INIT UNITS TSTPC
                                                CALL    SETDR           ;GET 1ST DRVS.
                                                CALL    RUN             ;DO TEST
                                                CALL    SWAPDR          ;GET NEXT DRVS.
                                                BCC     A               ;BR NO 2ND DRVS
                                                CALL    SETUP           ;REINIT UNITS TSTPC
                                                CALL    RUN             ;REPEAT TEST
                                A:                                      ;DONE
                                .NLIST
                                .NLIST ME
                                .LIST
                                .ENDM
                        ;------------------------------------------------------------------------
```

```
                                        .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
                                        ;--

005500  005002                  SWAPDR:: CLR     R2              ;FOR # OF DRIVE 1'S.
005502  012737  003340  005600          MOV     #BLKTBL,SWPTR   ;TABLE ADDR. OF 1ST UNIT
005510  017705  000064          1$:     MOV     @SWPTR,R5       ;GET DATA BLOCK ADDR.
005514  032715  100000                  BIT     #BIT15,@R5      ;ABORTED?
005520  001013                          BNE     3$              ;YES
005522  032765  000001  000060          BIT     #BIT0,DR(R5)    ;DID DR. 0?
005530  001007                          BNE     3$              ;NO, DID DR.1 1ST PASS
005532  032765  001000  000060          BIT     #BIT9,DR(R5)    ;YES; 1 SELECTED?
005540  001403                          BEQ     3$              ;NO, ALL DONE
005542  105265  000060                  INCB    DR(R5)          ;YES, SWAP
005546  005202                          INC     R2              ;ONE MORE TO TEST
005550  023727  005600  003356  3$:     CMP     SWPTR,#LSTDEV   ;LAST DEVICE?
005556  103004                          BHIS    4$              ;YES
005560  062737  000002  005600          ADD     #2,SWPTR        ;NO-POINT NEXT
005566  000750                          BR      1$              ;DO

005570  005702                  4$:     TST     R2              ;(CLEAR CARRY),MORE TO DO?
005572  001401                          BEQ     5$              ;NO
005574  000261                          SEC                     ;YES
005576  000207                  5$:     RETURN                  ;RETURN

005600  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
                                        ;--

005602  012737  003340  005656  SETDR:: MOV     #BLKTBL,SETPTR  ;TABLE OF ADDR. 1ST UNIT
005610  017705  000042          1$:     MOV     @SETPTR,R5      ;GET DATA BLOCK ADDR.
005614  105065  000060                  CLRB    DR(R5)          ;PRESET AS DR0
005620  032765  000400  000060          BIT     #BIT8,DR(R5)    ;DO DR0?
005626  001002                          BNE     2$              ;YES
005630  105265  000060                  INCB    DR(R5)          ;NO-USE DRIVE 1
005634  023727  005656  003356  2$:     CMP     SETPTR,#LSTDEV  ;MORE UNITS
005642  103004                          BHIS    3$              ;NO-EXIT
005644  062737  000002  005656          ADD     #2,SETPTR       ;YES-GET TABLE ENTRY
005652  000756                          BR      1$              ;CONFIGURE THAT UNIT
005654  000207                  3$:     RETURN
005656  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
                                        ;--

005660  012737  003340  005752  CLRALL:: MOV     #BLKTBL,CLRPTR  ;TOP OF TABLE OF ADDRESSES
005666  017705  000060          1$:     MOV     @CLRPTR,R5      ;GET DATA BLOCK
005672  004737  005720                  CALL    CLRBUF          ;CLEAR IT'S RECEIVE BUFFER
005676  023727  005752  003356          CMP     CLRPTR,#LSTDEV  ;LAST DEV?
005704  103004                          BHIS    2$              ;YES
005706  062737  000002  005752          ADD     #2,CLRPTR       ;-->NEXT
005714  000764                          BR      1$              ;CONTINUE
005716  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.
                                        ;--

005720                                  CLRBUF:: PUSH   R0              ;SAVE R0
005722                                           PUSH   R4              ;SAVE R4
005724  016500  000102                           MOV    RCVBUF(R5),R0   ;GET ADDRESS OF BUFFER
005730  012704  001036                           MOV    #RCBFSZ,R4      ;SIZE IN BYTES
005734  005020                          1$:      CLR    (R0)+           ;CLEAR IT
005736  162704  000002                           SUB    #2,R4           ;2 BYTES LESS
005742  001374                                   BNE    1$              ;MORE
005744                                           POP    R4              ;RESTORE
005746                                           POP    R0              ;
005750  000207                                   RETURN                 ;EXIT
005752  000000                          CLRPTR: .WORD
```

```
1553                                    ;++
1554                                    ; SETUP - CALLED WITHIN EACH TEST TO INSERT BEGINNING ADDRESS OF THE
1555                                    ;         TEST INTO ALL UNITS TEST PC'S.
1556                                    ; INPUTS:       TSTTOP LOADED WITH TEST ALGORITHMS STARTING ADDR.
1557                                    ;               BLKTBL - TOP OF DATA BLOCK ALLOCATION TABLE
1558                                    ;               LSTDEV - ADDR. OF LAST UNIT'S DATA BLOCK
1559                                    ;
1560                                    ; OUTPUTS:      TSTPC(R5) FOR ALL UNITS
1561                                    ;               DONE - CLEARED
1562                                    ;--
1563
1564 005754  005037  003314            SETUP:: CLR     DONE            ;NOT DONE YET
1565 005760  012737  003340  003316            MOV     #BLKTBL,IDPTR   ;TABLE TOP ADDR
1566 005766  017705  175324            1$:     MOV     @IDPTR,R5       ;DEVICE'S DATA BLOCK
1567 005772  013765  003320  000020            MOV     TSTTOP,TSTPC(R5);INSERT PC FOR TOP OF TEST
1568 006000  023727  003316  003356            CMP     IDPTR,#LSTDEV   ;ALL UNITS SET?
1569 006006  103004                           BHIS    2$              ;YES
1570 006010  062737  000002  003316           ADD     #2,IDPTR        ;NO,GET NEXT POINTER
1571 006016  000763                           BR      1$              ;SET HIM UP
1572 006020  000207                    2$:     RETURN                  ;DONE
```

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

006022  004737  006052          RUN::   CALL    NXTST           ;MAKE AND SEND NEXT PACK TO ALL
                                                                ;UNABORTED UNITS
006026  005737  003314                  TST     DONE            ;COMPLETE?
006032  001006                          BNE     2$              ;YES
006034  004737  006736                  CALL    GETANS          ;NO,GET ALL RESPONSES

006040                                  BREAK                   ;SUPERVISOR CHECK

006042  004737  010112                  CALL    CHKANS          ;CHECK ALL RESPONSES
006046  000765                          BR      RUN             ;CONTINUE TILL DONE
006050  000207                  2$:     RETURN
```

```
                                .SBTTL  NXTST / THE SCHEDULER
                                ;++
                                ; NXTST - USING EACH UN-ABORTED UNIT'S TEST PROGRAM COUNTER
                                ;         (TSTPC(R5)), EXECUTES THE TEST CODE THAT COMPRISES MAKING A
                                ;         PACKET AND SENDING IT.  ACTION IS ROUND ROBIN.  CHECKS FIRST
                                ;         FOR ANY UNIT RETRYING AND IF SO SERVICES ONLY THAT UNIT THIS
                                ;         PASS.  INITS NON-RETRYING UNITS IF NECESSARY.
                                ; 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.
                                ;                       SYSTAT IS UPDATED
                                ;--
003304  012737  003340  003304  NXTST:: MOV     #BLKTBL,DEVPTR  ;UNIT 0 TO START
006060  017705  175220          1$:     MOV     @DEVPTR,R5      ;GET DATA BLOCK
006064  032715  002000                  BIT     #BIT10,@R5      ;RETRYING?
006070  001422                          BEQ     2$              ;NOT THIS GUY
006072  005715                          TST     @R5             ;YES, ABORTED TOO?
006074  100420                          BMI     2$              ;YES ON TO NEXT UNIT
006076  052737  000002  003300          BIS     #BIT1,SYSTAT    ;NOT ABORTED-SET RETRY STATUS
006104                                  SWAPIN                  ;GET DEVICE REGISTERS
006130  004775  000020                  JSR     PC,@TSTPC(R5)   ;DO TEST FOR
006134  000477                          BR      NXTRET          ;THIS UNIT ONLY-EXIT
006136  023727  003304  003356  2$:     CMP     DEVPTR,#LSTDEV  ;TRY NEXT UNIT?
006144  103004                          BHIS    NXTST2          ;NO
006146  062737  000002  003304          ADD     #2.,DEVPTR      ;YES,->NEXT
006154  000741                          BR      1$              ;GET BLOCK

006156  005037  006336          NXTST2: CLR     ABONM           ;HERE=NO RETRIES TO DO, NO UNIT ABORTED YET
006162  012737  003340  003304          MOV     #BLKTBL,DEVPTR  ;-->UNIT 0 STORAGE BLOCK
006170  017705  175110          PERDEV: MOV     @DEVPTR,R5      ;R5-->NEXT DEVICE STORAGE BLOCK

006174  005715                  3$:     TST     @R5             ;ABORTED?
006176  100426                          BMI     4$              ;YES
006200  032715  040000                  BIT     #BIT14,@R5      ;SEND BREAK?
006204  001407                          BEQ     6$              ;NO
006206  004737  013222                  CALL    DOBRK           ;YES
006212  032715  040000                  BIT     #BIT14,@R5      ;SUCCESSFUL INIT?
006216  001016                          BNE     4$              ;NO ON TO NEXT UNIT
006220  005715                          TST     @R5             ;ABORTED?
006222  100414                          BMI     4$              ;YES-ON TO NEXT UNIT
006224                          6$:     SWAPIN                  ;NO,GET DEVICE REGISTERS R0-R4 CONTAINING TEST PARAMETERS
006250  004775  000020                  JSR     PC,@TSTPC(R5)   ;INITIATE 1 PACKET TRANSMISSION AND RETURN
006254  005715                  4$:     TST     @R5             ;ABORTED?
006256  100002                          BPL     8$              ;NO-ON TO NEXT UNIT
006260  005237  006336                  INC     ABONM           ;YES...ONE MORE TALLIED
006264  023727  003304  003356  8$:     CMP     DEVPTR,#LSTDEV  ;ALL TU'S TRIED?
006272  103004                          BHIS    5$              ;YES
006274  062737  000002  003304          ADD     #2.,DEVPTR      ;NO THE ADDRESS+2=NEXT ADDRESS
006302  000732                          BR      PERDEV          ;DO NEXT UNIT
006304  022737  000010  006336  5$:     CMP     #8.,ABONM       ;ALL ABORTED?
006312  001010                          BNE     NXTRET          ;NO
006314                                  ERRSF   1$0.,NOMOR      ;YES.
006324                          11$:    BREAK                   ;SUPERVISOR BREAK
```

```
1652 006326  005237  003332                    INC     ALLGON          ;SET DON'T-PRINT STATISTICS FLAG
1653 006332                                    DOCLN                   ;EXIT
1654 006334  000207                NXTRET:     RETURN
1655
1656 006336  000000                ABONM:      .WORD                   ;THE NUMBER OF ABORTED UNITS
1657 006340     101      114      114  NOMOR:  .ASCIZ  /ALL UNITS ABORTED!/
1658                                            .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.  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.
                        ; 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 -  OINTS 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.
                        ;--

006364  000240          RSVP::  NOP                     ;FINISH TEST
006366  012665  000020          MOV     (SP)+,TSTPC(R5) ;SAVE WHERE YOU WERE IN TEST BODY AND
006372                          SWAPOW                  ;SAVE TEST REGISTERS

                                                        ;CORRECT FOR RETURN TO SCHEDULER
006416  012700  024367  XFNSND: MOV     #TRBUF-1,R0     ;POINT TO XOFF
006422  005265  000070          INC     SNDCNT(R5)      ;ONE MORE TO SEND, TOO.
006426  000402                  BR      SND             ;SEND XOFF+PACKET
006430  012700  024370  NOXOFF: MOV     #TRBUF,R0       ;FOR NORMAL PACKET SEND
006434  004737  006666  SND:    CALL    SNDBYT          ;SEND BYTE
006440  005715                  TST     @R5             ;R5--> TO STATUS BLK
006442  100510                  BMI     6$              ;ABORTED? YES...QUIT
006444  005365  000070          DEC     SNDCNT(R5)      ;NO, SEND MORE
006450  001371                  BNE     SND             ;IF MORE TO SEND
006452  012700  024370          MOV     #TRBUF,R0       ;-- BUFFER
006456  016537  000064  003324  MOV     REC(R5),BLKER   ;PREPARE FOR RECEIVE
006464  156565  000032  000033  BISB    XSPKNM(R5),XSPKNM+1(R5)  ;REPLICATE LO. BYTE TO HI FOR GTPAKS, CMDAN.
006472  005065  000076          CLR     SUCCS(R5)       ;NO SUCCESS YET
006476  042715  001000          BIC     #BIT9,@R5       ;NO DATA CHK ERROR YET
006502  016565  000102  000104  MOV     RCVBUF(R5),PKPTR(R5)  ;TOP OF RCV BUFFER GOES THE IST PACKET
006510  012704  000034          MOV     #XSFLG,R4       ;FORM
006514  060504                  ADD     R5,R4           ;ADDRESS
006516  010465  000106          MOV     R4,XSPTR(R5)    ;OF 1ST XSFLG

006522  042715  000020          BIC     #BIT4,@R5       ;PRESET AS DATA PAK
```

```
1718 006526  121027  000002              CMPB    @R0,#RSCMND     ;WAS IT COMMAND PAK?
1719 006532  001054                      BNE     6$              ;NO...
1720 006534  116065  000002  000100      MOVB    2(R0),CMDSN1(R5)  ;YES-SAVE COMMAND
1721 006542  052715  000C20              BIS     #BIT4,@R5       ;ITS CMND PAK
1722
1723 006546  032715  002000              BIT     #BIT10,@R5      ;RETRYING?
1724 006552  001044                      BNE     6$              ;YES-DON'T UPDATE ANY STATS OR CONDITION
1725 006554  126027  000002  000002      CMPB    2(R0),#RSSRD    ;NO,A READ?
1726 006562  001012                      BNE     4$              ;NO
1727 006564  042715  000400              BIC     #BIT8,@R5       ;(FOR HARD/SOFT LOGGING) RD/WR FLAG-0
1728 006570  004737  013052              CALL    WHCHDR          ;GET DRIVE
1729 006574  103403                      BCS     8$              ;
1730 006576  005265  000114              INC     RDN0(R5)        ;DRIVE 0
1731 006602  000402                      BR      4$              ;
1732 006604  005265  000116      8$:     INC     RDN1(R5)        ;DRIVE 1
1733
1734 006610  126027  000002  000003  4$: CMPB    2(R0),#RSSWR    ;A WRITE?
1735 006616  001022                      BNE     6$              ;NO
1736 006620  052715  000400              BIS     #BIT8,@R5       ;YES, RD/WR FLAG=1
1737 006624  105760  000003              TSTB    3(R0)           ;VERIFY TOO?
1738 006630  001403                      BEQ     21$             ;NO
1739 006632  052715  000040              BIS     #BIT5,@R5       ;YES-SET VERIFY FLAG
1740 006636  000402                      BR      22$
1741 006640  042715  000040      21$:    BIC     #BIT5,@R5       ;(NO)-RESET VERIFY FLAG
1742 006644  004737  013052      22$:    CALL    WHCHDR          ;GET DRIVE NO
1743 006650  103403                      BCS     5$              ;CARRY-DR1
1744 006652  005265  000110              INC     WRTN0(R5)       ;# BLKS WRITTEN DR0
1745 006656  000402                      BR      6$              ;EXIT
1746
1747 006660  005265  000112      5$:     INC     WRTN1(R5)       ;# BLKS WRITTEN DRV'
1748 006664  000207              6$:     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
                                ;--

006666                          SNDBYT:: PUSH   R1              ;ENTER R0-->BYTE
006670  013701  003334          4$:     MOV     CSNRDY,R1       ;GET TIMEOUT CONSTANT FOR NOT READY ERROR
006674  105775  000026          1$:     TSTB    @XMSR(R5)       ;READY TO SEND?
006700  100412                          BMI     2$              ;YES
006702                                  PUSH    R0              ;NO, SAVE R0
006704                                  BREAK                   ;MONITOR BREAK
006706                                  POP     R0              ;RESTORE

006710  005301                          DEC     R1              ;ABORTED?
006712  001370                          BNE     1$              ;NO
006714  012704  000056                  MOV     #TOSNDB,R4      ;YES,SET CODE FOR TIMEOUT ERROR
006720  004737  012046                  CALL    LOG             ;LOG IT
006724  000402                          BR      3$              ;QUIT
006726  112075  000030          2$:     MOVB    (R0)+,@XMDB(R5) ;SEND IT
006732                          3$:     POP     R1              ;RESTORE
006734  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.
                                ;--

006736  000240                  GETANS:: NOP                    ;1 UNIT IF RETRY;  ELSE ALL
006740   32737  000002  003300          BIT     #BIT1,SYSTAT    ;RETRY?
006746  001010                          BNE     1$              ;YES
006750  012737  177777  007656          MOV     #-1,SERVST      ;PRESET NO UNITS SERVICED
006756  004737  005660                  CALL    CLRALL          ;CLEAR ALL INPUT BUFFERS
006762  004737  007214                  CALL    GTPKS8          ;GET ALL REPLYS
006766  000404                          BR      2$              ;EXIT
006770  004737  005720          1$:     CALL    CLRBUF          ;RETRY-CLEAR 1 UNIT ONLY
                                                                ;R5->UNIT BY NXTST
006774  004737  007004                  CALL    GTPKS1          ;GET 1 REPLY
007000  000207                  2$:     RETURN                  ;DONE

007002  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
                                ;--

007004  000240          GTPKS1:: NOP                     ;R5->THE UNIT
007006  012703  000034          MOV     #XSFLG,R3       ;THE OFFSET VALUE OF FLAG
007012  060503                  ADD     R5,R3           ;FORM THE ABSOLUTE ADDRESS
007014  010301                  MOV     R3,R1           ;R3-->ADDR. OF EXPECTED FLAG
007016  062701  000002          ADD     #2.,R1          ;R1-->ADDR. OF EXPECTED COUNT
007022  012700  007212          MOV     #EXON,R0        ;R0=ADDRESS
007026  004737  006666          CALL    SNDBYT          ;XON THE DEVICE
                                                        ;*** TIME CRITICAL
007032  016500  000102          MOV     RCVBUF(R5),R0   ;***--> TO THE BUFFER
007036  116502  000033          MOVB    XSPKNM+1(R5),R2 ;***GET THE # OF PACKETS TO RECEIVE
007042  032702  177400          BIT     #177400,R2      ;***SIGN UN-EXTEND
007046  011137  003310      1$: MOV     @R1,RCBCNT      ;***HOW MANY BYTES IT SHOULD BE
007052  011337  003306          MOV     @R3,RCFLG       ;***WHAT THE FIRST BYTE SHOULD BE
007056  004737  007662          CALL    GTBYTE          ;***GET THE ALL IMPORTANT FLAG
007062  032715  100000          BIT     #BIT15,@R5      ;TIMEOUT?
007066  001050                  BNE     4$              ;YES
007070  005300                  DEC     R0              ;-> BYTE RECIEVED
007072  111037  003301          MOVB    @R0,SYSTAT+1    ;SAVE IT AS FLAG BYTE
007076  121037  003306          CMPB    @R0,RCFLG       ;1ST BYTE WHAT WAS EXPECTED?
007102  001420                  BEQ     2$              ;YES
007104  121027  000002          CMPB    @R0,#RSEND      ;NO, WAS IT END PAK?
007110  001006                  BNE     14$             ;NO
007112  012737  000016  003310  MOV     #RSNDSZ,RCBCNT  ;YES, USE END SIZE FOR COUNT
007120  012702  000001          MOV     #1,R2           ;AND ASSUME IT'S LAST PACKET!
007124  000407                  BR      2$              ;CONTINUE RECEIVE
007126  121027  000001      14$: CMPB   @R0,#RSDATA     ;WAS IT DATA?
007132  001026                  BNE     4$              ;NO,CHKANS MAY FIND INIT...
007134  012737  000204  003310  MOV     #RSDASZ,RCBCNT  ;YES, SET FOR DATA PAK SIZE
007142  005202                  INC     R2              ;ONE MORE PACK THAN EXPECTED (END PACK)

007144  005200          2$: INC     R0                  ;RESTORE TO -> NEXT BYTE
007146  005337  003310      5$: DEC     RCBCNT          ;THAT'S ONE LESS BYTE TO GO
007152  001411                  BEQ     3$              ;DONE
007154  004737  007662          CALL    GTBYTE          ;GET NEXT OF PACKET
007160  005765  000074          TST     DLV(R5)         ;ERROR
007164  001011                  BNE     4$              ;YES-ALL OVER
007166  032715  100000          BIT     #BIT15,@R5      ;OR IF ABORTED
007172  001006                  BNE     4$              ;THEN QUIT
007174  000764                  BR      5$              ;CONTINUE RECEIVE

007176  005302          3$: DEC     R2                  ;ONE LESS PACKET TO GO
007200  001403                  BEQ     4$              ;MORE PACKETS IN TRANSACTION?
                                                        ;YES
```

```
1864 007202  022121                    CMP     (R1)+,(R1)+     ;POINT TO NEW EXPECTED COUNT
1865 007204  022323                    CMP     (R3)+,(R3)+     ;AND FLAG,
1866 007206  000717                    BR      $               ;AND RECEIVE,
1867 007210  000207              4$:   RETURN                  ;RETURN
1868
1869 007212     020              EXON:  .BYTE   $XON
1870 007213     023              EXOFF: .BYTE   RSXOFF
```

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

                                        ;++
                                        ; GTPKS8 - SET ALL ABORTED UNITS SERVICED (SERVST: BIT P' ITION).  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,
                                        ;          SEND "XOFF" TO ENHANCE THROUGHPUT AND GO ON TO NEXT UNIT
                                        ;          (IF ANY).
                                        ; 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
                                        ;--

007214  000240                  GTPKS8:: NOP                     ;GET ALL UNITS RESPONSES XOFF IF DATA PAK (THROUGHPUT)
007216  012737  003340  007660          MOV     #BLKTBL,GTPTR    ;->1ST
007224  017705  000430          GTAGIN: MOV     @GTPTR,R5        ;GET DATA BLOCK
007230  032715  100000                  BIT     #BIT15,@R5       ;ABORTED?
007234  001403                          BEQ     2$               ;NO
007236  004737  007572                  CALL    SETSRV           ;YES-SET' SERVICED' AND
007242  000534                          BR      GTDOWN           ;ON TO NEXT UNIT
007244  105765  000033          2$:     TSTB    XSPKNM+1(R5      ;NO, ANY PACKETS LEFT?
007250  001003                          BNE     3$               ;YES
007252  004737  007572                  CALL    SETSRV           ;NO-HE'S DONE
007256  000526                          BR      GTDOWN           ;SO ON TO NEXT UNIT
007260  105365  000033          3$:     DECB    XSPKNM+1(R5)     ;NOW ITS ONE LESS PACKET
007264  017537  000106  003306          MOV     @XSPTR(R5),RCFLG  ;GET EXPECTED FLAG
007272  062765  000002  000106          ADD     #2,XSPTR(R5)     ;--> COUNT
007300  017537  000106  003310          MOV     @XSPTR(R5),RCBCNT  ;AND EXPECTED COUNT
007306  012700  007212                  MOV     #EXON,R0         ;-> XON
                                                                 ;***TIME CRITICAL
007312  004737  006666                  CALL    SNDBYT           ;***SEND IT
007316  016500  000104                  MOV     PKPTR(R5),R0     ;***->WHERE 1ST BYTE GOES
007322  004737  007662                  CALL    GTBYTE           ;***GET IT
007326  032715  100000                  BIT     #BIT15,@R5       ;ABORTED?
007332  001403                          BEQ     4$               ;NO-CONTINUE
007334  105065  000033                  CLRB    XSPKNM+1(R5)     ;YES-NO MORE PACKETS EXPECTED
007340  000475                          BR      GTDOWN           ;ON TO NEXT
007342  005300                  4$:     DEC     R0               ;-->BYTE JUST RECEIVED
007344  111037  003301                  MOVB    @R0,SYSTAT+1     ;SAVE IT
007350  121037  003306                  CMPB    @R0,RCFLG        ;IS IT WHAT EXPECTED?
007354  001436                          BEQ     GTOK             ;YES
007356  105065  000033          UNXPCT: CLRB    XSPKNM+1(R5)     ;NO, MUST BE LAST REPLY
007362  121027  000002                  CMPB    @R0,#RSEND       ;MAYBE AN END PAK?
007366  001004                          BNE     4$               ;NO
007370  012737  000016  003310          MOV     #RSNDSZ,RCBCNT   ;YES, USE PROPER COUNT
007376  000406                          BR      GTUM             ;AND GET IT
007400  121027  000001          4$:     CMPB    @R0,#RSDATA      ;IS IT DATA?
007404  001053                          BNE     GTDOWN           ;NO, ALL OVER, CHKANS WILL INIT UNIT
007406  012737  000222  003310          MOV     #RSDNSZ,RCBCNT   ;YES, USE COUNT OF DATA + END PAK SURE TO FOLLOW
007414  005200                  GTUM:   INC     R0               ;WHERE TO STUFF THE REST
007416  005337  003310          5$:     DEC     RCBCNT           ;ONE DOWN
007422  001444                          BEQ     GTDOWN           ;NONE TO GO
007424  004737  007662                  CALL    GTBYTE           ;MORE TO GO
```

```
1930 007430  032715  100000            BIT     #BIT15,@R5      ;TIMEOUT?
1931 007434  001037                    BNE     GTDOWN          ;YES
1932 007436  005765  000074            TST     DLV(R5)         ;BUT DLV ERROR?
1933 007442  001765                    BEQ     5$              ;NO
1934 007444  105065  000033            CLRB    XSPKNM+1(R5)    ;YES-LAST TIME
1935 007450  000431                    BR      GTDOWN          ;ON TO NEXT
1936
1937 007452  005200            GTOK:   INC     R0              ;NEXT PLACE IN BUFFER
1938 007454  005337  003310      1$:   DEC     BYTCNT          ;MORE BYTES?
1939 007460  001413                    BEQ     2$              ;NO-ALL DONE
1940 007462  004737  007662            CALL    GTBYTE          ;YES-GET IT
1941 007466  032715  100000            BIT     #BIT15,@R5      ;TIMEOUT?
1942 007472  001020                    BNE     GTDOWN          ;YES
1943 007474  005765  000074            TST     DLV(R5)         ;ERROR?
1944 007500  001765                    BEQ     1$              ;NO
1945 007502  105065  000033            CLRB    XSPKNM+1(R5)    ;LAST TIME
1946 007506  000412                    BR      GTDOWN          ;EXIT
1947 007510  122775  000001  000104  2$:   CMPB    #RSDATA,@PKPTR(R5)  ;WAS DATA?
1948 007516  001006                    BNE     GTDOWN          ;NO, ALL DONE
1949 007520  010065  000104            MOV     R0,PKPTR(R5)    ;START OF NEXT PACK NEXT TIME
1950 007524  012700  007213            MOV     #EXOFF,R0       ;XOFF AND SEND TO
1951 007530  004737  006666            CALL    SNDBYT          ;ENHANCE THROUGHPUT
1952 007534  062765  000002  000106  GTDOWN: ADD     #2.,XSPTR(R5)   ;NEXT XSFLG FOR NEXT TRY
1953 007542  023727  007660  003356        CMP     GTPTR,#LSTDEV   ;DONE ONE CYCLE ALL UNITS?
1954 007 50  103004                    BHIS    1$              ;YES
1955 007552  062737  000002  007660        ADD     #2,GTPTR        ;NEXT UNIT
1956 007560  000621                    BR      GTAGIN          ;CONTINUE RECEIVE
1957 007562  105737  00765e          1$:   TSTB    SERVST          ;DONE SERVICING ALL PAKS
1958                                                            ;FROM ALL UNITS?
1959 007566  001212                    BNE     GTPKS8          ;NO, KEEP TRYING
1960 007570  000207                    RETURN                  ;YES.
```

```
                                .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
                                ;--

007572                          SETSRV: PUSH    R5              ;SET UNIT SERVICED
007574                                  PUSH    R0
007576  011505                          MOV     @R5,R5          ;GET STAT WD
007600  042705  177770                  BIC     #177770,R5      ;MASK UNIT #
007604  012700  007636                  MOV     #SRVTBL,R0      ;->TOP OF BIT TABLE
007610  005705                  1$:     TST     R5              ;RIGHT ONE?
007612  001404                          BEQ     2$              ;YES
007614  062700  000002                  ADD     #2,R0           ;NO, ->NEXT
007620  005305                          DEC     R5              ;1 LESS
007622  000772                          BR      1$              ;CONTINUE
007624  041037  007656          2$:     BIC     @R0,SERVST      ;MOW IT DOWN
007630                                  POP     R0
007632                                  POP     R5
007634  000207                          RETURN                  ;RETURN

007636  000001                  SRVTBL: .WORD   BIT0            ;BIT POSITION LOOKUP TABLE
007640  000002                          .WORD   BIT1
007642  000004                          .WORD   BIT2
007644  000010                          .WORD   BIT3
007646  000020                          .WORD   BIT4
007650  000040                          .WORD   BIT5
007652  000100                          .WORD   BIT6
007654  000200                          .WORD   BIT7

007656  000000                  SERVST: .WORD
007660  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
                                ;--

007662  005037  010106          GTBYTE:: CLR     GBTMP           ;TIMEOUT REGISTER
007666  013704  003334                   MOV     CSRCVB,R4       ;TIMEOUT ERROR CONSTANT (MULTIPLIER)
007672  105775  000022          1$:      TSTB    @RCSR(R5)       ;READY?
007676  100013                           BPL     3$              ;NO
007700  017565  000024  000074           MOV     @RCDB(R5),DLV(R5)  ;GET ERROR + BYTE
007706  116520  000074                   MOVB    DLV(R5),(R0)+   ;COPY BYTE TO BUFFER
007712  005765  000074                   TST     DLV(R5)         ;ERROR?
007716  100472                           BMI     4$              ;YES-EXIT
007720  005065  000074                   CLR     DLV(R5)         ;NO-RESET
007724  000467                           BR      4$              ;AND EXIT
007726  005337  010106          3$:      DEC     GBTMP           ;DEC T.O. CONSTANT
007732  001357                           BNE     1$              ;STILL VALID

                                ;CODE TO SEE ^C DURING LONG SEEK OR REWIND
                                                                 ;HERE GBTMP=0
007734  010037  010110                   MOV     R0,GBTMP2       ;R0 MUST BE PRESERVED!
007740  012700  007213                   MOV     #EXOFF,R0       ;QUIET THE DEVICE
007744  004737  006666                   CALL    SNDBYT          ;BY SENDING XOFF
007750  105775  000022          6$:      TSTB    @RCSR(R5)       ;CHARACTER SLOP OVER?
007754  100415                           BMI     5$              ;YES
007756  005337  010106                   DEC     GBTMP           ;NO-WAIT A WHILE
007762  105737  010106                   TSTB    GBTMP           ;DONE WAITING?
007766  001370                           BNE     6$              ;NO
007770                                   BREAK                   ;YES-NO SLOP OVER
007772  012700  007212                   MOV     #EXON,R0        ;START DEVICE TALKING
007776  004737  006666                   CALL    SNDBYT          ;AGAIN
010002  013700  010110                   MOV     GBTMP2,R0       ;RESTORE R0
010006  000426                           BR      7$              ;END KLUGE
010010  013700  010110          5$:      MOV     GBTMP2,R0       ;RESTORE R0
010014  017565  000024  000074           MOV     @RCDB(R5),DLV(R5)  ;GET ERROR + BYTE
010022  116520  000074                   MOVB    DLV(R5),(R0)+   ;COPY BYTE TO BUFFER
010026  005765  000074                   TST     DLV(R5)         ;ERROR?
010032  100403                           BMI     17$             ;YES-EXIT
010034  005065  000074                   CLR     DLV(R5)         ;NO-CLEAR
```

```
 2057 010040  000400                      BR      17$             ;EXIT
 2058 010042  010037  010110      17$:    MOV     R0,GBTMP2       ;AGAIN SAVE R0
 2059 010046  012700  007212              MOV     #EXON,R0        ;RESTORE TO TALKING STATE
 2060 010052  004737  006666              CALL    SNDBY'          ;BY SENDING 'XON'
 2061 010056  013700  010110              MOV     GBTMP2,R0       ;RESTORE R0
 2062 010062  000410                      BR      4$              ;DONE
 2063 010064  005037  010106      7$:     CLR      GBTMP
 2064 010070  005304                      DEC     R4              ;TIMEOUT?
 2065 010072  001277                      BNE     1$              ;NO
 2066 010074  012704  000050              MOV     #TORCVB,R4      ;YES
 2067 010100  004737  012046              CALL    LOG             ;LOG ERROR.
 2068 010104  000207              4$:     RETURN                  ;RETURN
 2069 010106  000000              GBTMP:  .WORD   0
 2070 010110  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.
                                ; INPUTS:  IMPLIED SYSTAT BIT1 (RETRYING)
                                ;          BLKTBL - TOP OF DATA BLOCK ALLOCATION TABLE
                                ;          LSTDEV - ADDR. OF LAST UNIT'S DATA BLOCK
                                ;
                                ; OUTPUTS: NONE PASSED.
                                ;--

010112  000240                  CHKANS:: NOP                    ;IF RETRY THEN CHECK ONE
                                                                ;ELSE CHECK ALL
010114  032737  000002  003300          BIT     #BIT1,SYSTAT    ;RETRYING?
010122  001403                  BEQ     CHK8                    ;NO DO NORMAL
010124  004737  010202          CALL    CHKPKS                  ;YES DO BAZARRE WITH
                                                                ;R5 -> UNIT
010130  000422                  BR      CHKANR                  ;ALL DONE

010132  012737  003340  010200  CHK8:   MOV     #BLKTBL,CHKPTR  ;YOU KNOW ... TOP OF TABLE
010140  017705  000034          2$:     MOV     @CHKPTR,R5      ;GET UNIT'S BLOCK ADDRESS
010144  032715  100000                  BIT     #BIT15,@R5      ;ABORTED?
010150  001002                  BNE     3$                      ;YES
010152  004737  010202          CALL    CHKPKS                  ;NO, DO THIS GUY
010156  023727  010200  003356  3$:     CMP     CHKPTR,#LSTDEV  ;ALL DONE?
010164  103004                  BHIS    CHKANR                  ;YES
010166  062737  000002  010200          ADD     #2,CHKPTR       ;NO,-->NEXT DEVICE
010174  000761                  BR      2$                      ;DO DA

010176  000207                  CHKANR: RETURN

010200  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.
                        ; INPUTS:  (IMPLIED) UNITS DATA BLOCKS
                        ;
                        ; OUTPUTS: ERRORS - DLV ERROR
                        ;                  - UNKNOWN FLAG BYTE ERROR
                        ;                  - CHECKSUM ERROR
                        ;                  - DATA COMPARE ERROR
                        ;          R4 = ERROR NUMBER
                        ;          SYSTAT UPPER BYTE = 1ST BYTE OF RESPONSE
                        ;--

010202  000240          CHKPKS:: NOP                    ;CHECK WHAT WAS RECIEVED
010204  016500  000102          MOV     RCVBUF(R5),R0   ;GET BUFFER ADDR.
010210  016502  000032          MOV     XSPKNM(R5),R2   ;AND # OF PACKETS EXPECTED
010214  012703  000034          MOV     #XSFLG,R3       ;THE OFFSET VALUE
010220  060503                  ADD     R5,R3           ;R3-->THIS UNIT XSFLG AGAIN
010222  010301                  MOV     R3,R1           ;COPY TO R1
010224  062701  000002          ADD     #2,R1           ;R1-->XSBCNT FOR 1ST PACKET
010230  010065  000104  1$:     MOV     R0,PKPTR(R5     ;POINT TO PACKET
010234  111037  003301          MOVB    @R0,SYSTAT+1    ;SAVE RCV'D BYTE
010240  011137  003310          MOV     @R1 RCBCNT      ;GET COUNT
010244  011337  003306          MOV     @R3,RCFLG       ;AND FLAG
010250  121013                  CMPB    @R0,@R3         ;1ST BYTE=EXPECTED?
010252  001050                  BNE     5$              ;UH OH...
010254  121027  000020          CMPB    @R0,#RSCONT     ;OK, IS IT 1 BYTE?
010260  001516                  BEQ     7$              ;YES...ONTO NEXT PACK
                                                        ;NO, SO > 1 BYTE (NEVER EXPECT INIT!)
010262  013704  003310          MOV     RCBCNT,R4       ;EXPECTED, SO COUNT MUST BE RIGHT
010266  005744                  TST     -(R4)           ;ADJUST FROM RECEIVE COUNT TO COUNT FOR CHECKSUM
010270  004737  013162          CALL    CKCKSM          ;CHECK CHECKSUM
010274  103005                  BCC     2$              ;NO CARRY...NO INCORRECT
010276  012704  000022          MOV     #BDCHK,R4       ;ERROR
010302  004737  012046          CALL    LOG             ;LOG IT
010306  000503                  BR      7$              ;ON TO NEXT PACK
010310  122710  000002  2$:     CMPB    #RSEND,(R0)     ;END PAK?
010314  001005                  BNE     3$              ;NO
010316  004737  010536          CALL    CHKEND          ;YES-CHECK
010322  012702  000001          MOV     #1,R2           ;LAST PACKET
010326  000473                  BR      7$              ;AND FALL THROUGH
010330  122710  000001  3$:     CMPB    #RSDATA,@R0     ;DATA PAK?
010334  001003                  BNE     4$              ;NO
010336  004737  013762          CALL    COMPAR          ;YES-CHECK DATA
010342  000465                  BR      7$              ;ALL DONE?

010344  052715  040000  4$:     BIS     #BIT14,@R5      ;SET 'DOBREAK' FLAG
010350  012704  000052          MOV     #OTL,R4         ;OUT TO LUNCH
010354  005765  000074          TST     DLV(R5)         ;AH,BUT DLV ERROR?
010360  001402                  BEQ     20$             ;NO
010362  012704  000012          MOV     #OVRN,R4        ;YES-USE CORRECT ERROR #
010366  004737  012046  20$:     CALL    LOG             ;TALLY
```

```
2165 010372  000460                        BR      8$              ;DONE
2166
2167                                ;HERE CHECKS UNEXPECTED RESPONSE
2168
2169 010374  122710  000004        5$:     CMPB    #RSINIT,@R0     ;INIT?
2170 010400  001007                        BNE     6$              ;NO
2171 010402  052715  040000                BIS     #BIT14,@R5      ;YES-SET 'DOBREAK' FLAG
2172 010406  012704  000006                MOV     #RCINIT,R4      ; WE GOT AN INIT
2173 010412  004737  012046                CALL    LOG             ;TALLY IT
2174 010416  000446                        BR      8$              ;DONE
2175 010420  122710  000001        6$:     CMPB    #RSDATA,@R0     ;DATA PAK?
2176 010424  001013                        BNE     9$              ;NO
2177 010426  012704  000204                MOV     #RSDASZ,R4      ;YES, USE DATA SIZE
2178 010432  005744                        TST     -(R4)           ;ADJUST FOR CHKSUM
2179 010434  004737  013162                CALL    CKCKSM          ;AND CHECK
2180 010440  103421                        BCS     10$             ;GOOF
2181 010442  004737  013762                CALL    COMPAR          ;OK, HOW'S THE DATA?
2182                                                               ;EXPECTED END, GOT
2183                                                               ;DATA + END.
2184 010446  062700  000204                ADD     #RSDASZ,R0      ;POINT TO END PACK
2185 010452  000666                        BR      1$              ;CHECK IT, USE SAME XSFLG
2186
2187 010454  122710  000002        9$:     CMPB    #RSEND,(R0)     ;END?
2188 010460  001331                        BNE     4$              ;NO-OUT TO LUNCH
2189
2190 010462  012704  000016                MOV     #RSSNSZ,R4      ;YES, TOTAL SIZE MINUS
2191 010466  005744                        TST     -(R4)           ;TWO (THE CHKSUM)
2192 010470  004737  013162                CALL    CKCKSM          ;CHECK IT
2193 010474  103403                        BCS     10$             ;OOPS
2194 010476  004737  010536                CALL    CHKEND          ;OK,NOW TEST SUC. CODE
2195
2196 010502  000414                        BR      8$              ;ALL DONE
2197
2198 010504  012704  000022        10$:    MOV     #BDCHK,R4       ;CHECKSUM ERROR
2199 010510  004737  012046                CALL    LOG
2200 010514  000407                        BR      8$              ;EXIT
2201
2202 010516  005302                7$:     DEC     R2              ;ANY PACKETS LEFT TO CHECK?
2203 010520  001405                        BEQ     8$              ;NO, ALL DONE
2204 010522  063700  003310                ADD     RCBCNT,R0       ;YES, POINT TO NEXT PACKET
2205 010526  022121                        CMP     (R1)+,(R1)+     ;POINT TO NEXT EXPECTED COUNT
2206 010530  022323                        CMP     (R3)+,(R3)+     ;AND EXPECTED FLAG
2207 010532  000636                        BR      1$              ;TRY ANOTHER,THEY'RE SMALL
2208
2209 010534  000207                8$:     RETURN                  ;RETURN
```

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

                                        ;++
                                        ; CHKEND - IF RETRYING; 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;  CHECK IF 'DATA CHECK' ERROR SUCCESS CODE,
                                        ;          AND IF SO,START RETRY, ELSE EXIT.
                                        ; INPUTS:  IMPLIED  UNITS DATA BLOCK
                                        ; OUTPUTS: RETRY (SYSTAT BIT 1), (BIT10 @R5) SET IF RETRYING.
                                        ;          -  DATA COMARE ERROR (BIT6 @R5) CLEARED.
                                        ;          -   REDUCED/NORMAL GAIN (BIT7 @R5) ADJUSTED
                                        ;--

010536                                  CHKEND:: PUSH   R0              ;R0 --> END PAK
010540                                           PUSH   R4
010542  032737  000002  003300          1$:      BIT    #BIT1,SYSTAT    ;RETRYING?
010550  001052                                   BNE    CHKREE          ;YES-BRANCH
010552  004737  011536                           CALL   CHKSUC          ;NO,GET SUCCESS CODE
                                                                        ;LOG ERROR...
010556  032715  100000                           BIT    #BIT15,@R5      ;ABORTED?
010562  001402                                   BEQ    3$              ;NO,CONTINUE
010564  000137  011242                           JMP    CHKRET          ;YES,EXIT
010570  105765  000077                  3$:      TSTB   SUCCS+1(R5)     ;NO; HOW'D WE DO?
010574  001013                                   BNE    CHKERR          ;NOT SO GOOD.
010576  032715  000100                           BIT    #BIT6,@R5       ;OK, HOST FIND DATA PAK ERROR?
010602  001002                                   BNE    2$              ;YES
010604  000137  011242                           JMP    CHKRET          ;NO
010610  012704  000014                  2$:      MOV    #BDCOM,R4       ;YES;  JUST BAD DATA-NO DATACHK ERR
010614  004737  012046                           CALL   LOG             ;BAD DATA IN PACKET
010620  000137  011242                           JMP    CHKRET          ;QUIT
010624  032715  001000                  CHKERR:  BIT    #BIT9,@R5       ;BAD SUCCESS; TU DATA CHK ERROR?
010630  001002                                   BNE    1$              ;YES
010632  000137  011242                           JMP    CHKRET          ;NO. ALL DONE.
010636  052715  002000                  1$:      BIS    #BIT10,@R5      ;YES-START RETRY
010642  012765  000001  000002                   MOV    #1,RETRY(R5)    ;CALL IT 1ST
010650                                           PRINTX #RTRYN,RETRY(R5)  ;** PRINT **
010674  000562                                   BR     CHKRET          ;ALL DONE
010676  004737  011536                  CHKREE:  CALL   CHKSUC  ;RETRYING,GET SUCCESS
010702  105765  000077                           TSTB   SUCCS+1(R5)     ; SUCCESSFUL YET?
010706  001054                                   BNE    UNSUC           ;NO, CHECK COUNT
010710                                           PRINTX #RECOV,RETRY(R5)
010734  105715                                   TSTB   (R5)            ;DETERMINE THRESHOLD
010736  100411                                   BMI    2$              ;IT'S MODIFIED
010740                                           PRINTX #THRSLO         ;NORMAL
010760  000410                                   BR     3$
010762                                  2$:      PRINTX #THRSHI         ;ENHANCED
011002  032715  000400                  3$:      BIT    #BIT8,@R5       ;WRITE OR READ OPERATION?
011006  001003                                   BNE    4$              ;WRITE
011010  012704  000002                           MOV    #SFTRD,R4       ;READ
011014  000402                                   BR     5$
011016  012704  000004                  4$:      MOV    #SFTWR,R4       ;WRITE
011022  004737  012046                  5$:      CALL   LOG
011026  005065  000002                           CLR    RETRY(R5)       ;RESTORE TO NORMAL STATE
011032  042715  002200                           BIC    #BIT10!BIT7,@R5 ;NO RETRY, NORM THRESHOLD
```

```
2269 011036  000501                          BR      CHKRET          ;QUIT
2270
2271 011040  000240                  UNSUC:  NOP                     ;RETRYING;  SEE IF HARD YET
2272 011042  032715  001000                  BIT     #BIT9,@R5       ;TU DATA CHECK ERROR?
2273 011046  001015                          BNE     2$              ;YES
2274 011050                                  PRINTB  #RETERR         ;NO-"OTHER-ERROR" ERROR
2275 011070  005065  000002                  CLR     RETRY(R5)       ;NO RETRIES
2276 011074  042715  002200                  BIC     #BIT10!BIT7,@R5 ;NO RETRY, NORM THRESHOLD
2277 011100  000460                          BR      CHKRET          ;EXIT
2278 011102  023765  003322  000002  2$:     CMP     MXRTRY,RETRY(R5)  ;YES. DID WE GRADUATE TO HARD?
2279 011110  001425                          BEQ     HRD1            ;YES
2280 011112  005265  000002                  INC     RETRY(R5)       ;NO.  JUST ANOTHER
2281 011116                                  PRINTX  #RTRYN,RETRY(R5)  ;PRINT OUT
2282 011142  032715  000200                  BIT     #BIT7,@R5       ;WAS NORMAL THRESHOLD?
2283 011146  001403                          BEQ     1$              ;YES-REDUCE GAIN
2284 011150  042715  000200                  BIC     #BIT7,@R5       ;NO-NORM
2285 011154  000432                          BR      CHKRET
2286 011156  052715  000200          1$:     BIS     #BIT7,@R5       ;REDUCED
2287 011162  000427                          BR      CHKRET          ;DONE
2288 011164  000240                  HRD1:   NOP                     ;HERE IS HARD ERROR!
2289 011166                                  PRINTX  #UNREC
2290 011206  032715  000400                  BIT     #BIT8,@R5       ;RD OR WR?
2291 011212  001003                          BNE     4$              ;WRITE
2292 011214  012704  000016                  MOV     #HRDRD,R4       ;READ
2293 011220  000402                          BR      5$              ;LOG IT
2294 011222  012704  000020          4$:     MOV     #HRDWR,R4       ;WRITE
2295 011226  004737  012046          5$:     CALL    LOG             ;LOG IT
2296 011232  005065  000002                  CLR     RETRY(R5)       ;BACK TO NORMAL
2297 011236  042715  002200                  BIC     #BIT10!BIT7,@R5 ;NO RETRY, NOT REDUCED
2298
2299 011242  042737  000002  003300  CHKRET: BIC     #BIT1,SYSTAT    ;NO SYSTEM RETRY NEXT PASS
2300 011250  042715  000100                  BIC     #BIT6,@R5       ;NO MORE HOST DATA CHECK ERROR
2301 011254                                  POP     R4
2302 011256                                  POP     R0
2303 011260  000207                          RETURN
2304
2305
2306 011262      045     101     122  RECOV:  .ASCIZ  /%ARECOVERED FROM DATA CHECK ERROR RETRY # %D1%N/
2307                                          .EVEN
2308 011342      045     101     040  THRSLO: .ASCIZ  /%A NORMAL THRESHOLD%N/
2309                                          .EVEN
2310 011370      045     101     040  THRSHI: .ASCIZ  /%A MODIFIED THRESHOLD %N/
2311                                          .EVEN
2312 011422      045     101     122  RTRYN:  .ASCIZ  /%ARETRY # %D1%N/
2313                                          .EVEN
2314 011442      045     101     125  UNREC:  .ASCIZ  /%AUNRECOVERABLE%N/
2315                                          .EVEN
2316 011464      045     101     117  RETERR: .ASCIZ  /%AOTHER ERROR DURING RETRY : EXIT RETRY%N/
2317                                          .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
                                ;--

011536  000240          CHKSUC:: NOP
011540  016065  000002  000076          MOV     2(R0),SUCCS(R5) ;R0-->END PACKET
011546  122760  000000  000003          CMPB    #ESOK,3(R0,     ;GET SUCCESS BYTE
011554  001533                          BEQ     12$             ;COMPLETE SUCCESS-EXIT

011556  122760  000001  000003          CMPB    #ESTRY,3(R0)    ;OK BUT RETRIES?
011564  001012                          BNE     20$             ;NO
011566  126527  000100  000002          CMPB    CMDSNT(R5),#RSSRD;A READ?
011574  001001                          BNE     22$             ;NO

011576  000516                          BR      10$             ;NO RETRIES IN MAINTENANCE!
011600  126527  000100  000003  22$:    CMPB    CMDSNT(R5),#RSSWR       ;A WRITE?
011606  001001                          BNE     20$             ;NO
011610  000511                          BR      10$             ;LOG IT
011612  122760  177737  000003  20$.    CMPB    #ESNOMO,3(R0)   ;NO MOTOR?
011620  001003                          BNE     1$              ;NO
011622  012704  000030                  MOV     #NOMOT,R4       ;YES-
011626  000504                          BR      11$             ;LOG

011630  122760  177757  000003  1$:     CMPB    #ESCKS,3(R0)    ;"DATA CHECK" ERROR?
011636  001003                          BNE     2$              ;NO
011640  052715  001000                  BIS     #BIT9,@R5       ;SET DATA-CHK-ERROR FLAG
011644  000477                          BR      12$             ;DONT LOG

011646  126527  000100  000007  2$:     CMPB    CMDSNT(R5),#RSSSLF  ;SELF TEST?
011654  001006                          BNE     3$              ;NOPE
011656  105760  000003                  TSTB    3(R0)           ;YES, NEG. IF ERROR
011662  100070                          BPL     12$             ;OK

011664  012704  000044                  MOV     #SLFER,R4       ;YES-ERROR
011670  000463                          BR      11$             ;LOG IT

011672  122760  177740  000003  3$:     CMPB    #ESSK,3(R0)     ;SEEK ERROR?
011700  001003                          BNE     4$              ;NO
011702  012704  000024                  MOV     #SKERR,R4       ;YES-
011706  000454                          BR      11$             ;LOG

011710  122760  177767  000003  4$:     CMPB    #ESNCRT,3(R0)   ;NO CART?
011716  001003                          BNE     5$              ;NO
011720  012704  000054                  MOV     #NCART,R4       ;YES-
011724  000445                          BR      11$             ;LOG

011726  122760  177720  000003  5$:     CMPB    #ESCMD,3(R0)    ;NO UNDERSTAND HOST?
```

```
2377 011734  001003                        BNE     6$              ;NO
2378 011736  012704  000040                MOV     #CMNDER,R4      ;YES-
2379 011742  000436                        BR      11$             ;LOG
2380
2381 011744  122760  177770  000003  6$:   CMPB    #ESNONX,3(R0)   ,NON EXISTENT UNIT?
2382 011752  001003                        BNE     7$              ;NO
2383 011754  012704  000036                MOV     #NOUNIT,R4      ;YES-
2384 011760  000427                        BR      11$             ;LOG
2385
2386 011762  122760  177765  000003  7$:   CMPB    #ESWLOC,3(R0)   ;WRITE LOCKED?
2387 011770  001003                        BNE     8$              ;NO
2388 011772  012704  000026                MOV     #WRLOCK,R4      ;YES-
2389 011776  000420                        BR      11$             ;LOG
2390
2391 012000  122760  177776  000003  8$:   CMPB    #ESPART,3(R0)   ;PARTIAL OP?
2392 012006  001003                        BNE     9$              ;NO
2393 012010  012704  000034                MOV     #PARTL,R4       ;YES-
2394 012014  000411                        BR      11$             ;LOG
2395
2396 012016  122760  177711  000003  9$:   CMPB    #ESREC,3(R0)    ;WRONG RECORD?
2397 012024  001003                        BNE     10$             ;NO
2398 012026  012704  000042                MOV     #RECERR,R4      ;YES-
2399 012032  000402                        BR      11$             ;LOG
2400
2401 012034  012704  000046          10$:  MOV     #SUCOTL,R4      ;UNDEFINED
2402 012040  004737  012046          11$:  CALL    LOG             ;LOG ERROR
2403 012044  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
                                ;--

012046                          LOG::   PUSH    R0
012050                                  PUSH    R1
012052                                  PUSH    R3
012054                                  PUSH    R4

012056  011537  002074                  MOV     @R5,L$LUN       ;GET UNIT NUMBER
012062  042737  177770  002074          BIC     #177770,L$LUN   ;MASK IT OFF
012070  010465  000004                  MOV     R4,ABNDX(R5)    ;SAVE INDEX IN CASE OF ABORT MESSAGE
012074  012703  000120                  MOV     #LGOFST,R3      ;OFFSET TO LOW ORDER BYTE (DRIVE0)
012100  060403                          ADD     R4,R3           ;FORM INDEX OF PARAM. TO UPDATE
012102  060503                          ADD     R5,R3           ;FORM ABSOLUTE ADDR. THIS UNIT
012104  004737  013052                  CALL    WHCHDR          ;SEE WHICH DRIVE T'WAS
012110  103001                          BCC     2$              ;WAS DRIVE 0
012112  005203                          INC     R3              ;DRIVE 1; POINT TO UPPER BYTE
012114  122713  000377          2$:     CMPB    #255.,@R3       ;POTENTIAL OVERFLOW POSSIBLE?
012120  001005                          BNE     LOGOK           ;NO
012122                          LOGO:   ERRDF   0.,OVRFLO,ERRDES  ;YES
012132  000512                          BR      ABO             ;ABORT UNIT
012134  105213                  LOGOK:  INCB    @R3             ;INCREMENT THE ERROR
012136  111304                          MOVB    @R3,R4          ;TEMP'LY SAVE IT
012140  016503  000004                  MOV     ABNDX(R5),R3    ;GET INDEX AGAIN
012144  012701  002220                  MOV     #RSNTAB,R1      ;FORM ADRS OF MSG
012150  066501  000004                  ADD     ABNDX(R5),R1    ;LIKE THIS
012154  042701  000001                  BIC     #B!TO,R1        ;INSURE WORD BOUNDARY
012160  032737  000004  016154          BIT     #EVL,FLGLOC     ;EVL SELECTED?
012166  001414                          BEQ     LOGOK2          ;NO-CONT
012170  123704  002216                  CMPB    EVLTHR,R4       ;YES,OVER THRESHOLD?
012174  101011                          BHI     LOGOK2          ;NO
012176  010337  012210                  MOV     R3,DFTL1+2      ;YES,LOAD ERROR #
012202  011137  012212                  MOV     @R1,DFTL1+4     ;AND MESSAGE ADDR
012206                          DFTL1:  ERRDF   0,DFTL1,ERRDES  ;ERROR
012216  000460                          BR      ABO             ;DROP IT
012220  120327  000014          LOGOK2: CMPB    R3,#BDCOM       ;'NEVER FATAL' TYPE?
012224  103011                          BHIS    NTSFT           ;NO
012226  010337  012240                  MOV     R3,LOG1+2       ;YES, ERROR CODE
012232  011137  012242                  MOV     @R1,LOG1+4      ;DESCRIPTION
012236                          LOG1:   ERRSOFT 0.,LOG1,ERRDES
012246  000450                          BR      LOGO            ;EXIT

012250  120327  000026          NTSFT:  CMPB    R3,#WRLOCK      ;ONE TRY?
012254  103411                          BLO     MABEE           ;NO, MAYBE A MULTIPLE
012256  010337  012270                  MOV     R3,LOG2+2.      ;YES
```

```
2463 012262  011137  012272                    MOV     @R1,LOG2+4
2464 012266                            LOG2:   ERRHRD  0,LOG2,ERRDES   ;PRINT HARD MESSAGE
2465 012276  000430                            BR      ABO             ;DROP UNIT
2466
2467 012300  042704  177400            MABEE:  BIC     #177400,R4      ;NEGATE SIGN EXTEND
2468 012304  163704  003312            1$:     SUB     FTLNM,R4        ;SEE IF MULTIPLE OF
2469 012310  001413                            BEQ     HRD             ;FTLNM-YES!
2470 012312  103401                            BLO     SFT             ;NO
2471 012314  000773                            BR      1$              ;NOT THERE YET
2472
2473 012316  010337  012330            SFT:    MOV     R3,LOG3+2       ;ERROR CODE
2474 012322  011137  012332                    MOV     @R1,LOG3+4      ;DESCRIPTION
2475 012326                            LOG3:   ERRSOFT 0,LOG3,ERRDES
2476 012336  000414                            BR      LOGO            ;EXIT
2477 012340  010337  012352            HRD:    MOV     R3,LOG3B+2      ;HARD ERROR CODE
2478 012344  011137  012354                    MOV     @R1,LOG3B+4     ;DESCRIPTION
2479 012350                            LOG3B:  ERRHRD  0,LOG3B,ERRDES
2480
2481 012360  011500                    ABO:    MOV     @R5,R0          ;GET UNIT NUMBER
2482 012362  042700  177770                    BIC     #177770,R0      ;UN-SIGN EXTEND
2483 012366                                    DODU    R0              ;USE LOGICAL # TO DROP
2484 012370                            LOGO:   POP     R4              ;RESTORE
2485 012372                                    POP     R3
2486 012374                                    POP     R1
2487 012376                                    POP     R0
2488 012400  000207                            RETURN                  ;RETURN
```

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

012402                                          BGNMSG  ERRDES          ;ERROR DESCRIPTION
012402                                          PUSH    R0
012404                                          PUSH    R2
012406  005002                                  CLR     R2              ;PRESET TO DATA TYPE
012410  032715  000020                          BIT     #BIT4,@R5       ;WHAT PACK TYPE?
012414  001401                                  BEQ     2$              ;DATA
012416  005202                                  INC     R2              ;COMMAND
012420                                  2$:     PRINTB  #UNIT,<B,DR(R5)>,R2,<B,SYSTAT+1>
012456  016500  000064                          MOV     REC(R5),R0      ;RECORD NUMBER
012462  016502  000072                          MOV     PATTEN(R5),R2   ;DATA EXPECTED
012466                                          PRINTB  #RECID,R0,<B,CMDSNT(R5)>,<B,R2>,<B,SUCCS+1(R5)>
012530  005765  000074                          TST     DLV(R5)         ;DLV ERROR?
012534  001414                                  BEQ     3$              ;NO
012536                                          PRINTB  #RECID2,DLV(R5) ;YES-PRINT
012562  005065  000074                          CLR     DLV(R5)         ;RESET
012566                                  3$:     POP     R2              ;RESTORE
012570                                          POP     R0
012572                                          ENDMSG                  ;EXIT
012574      045     101     104  UNIT::  .ASCIZ  /%ADRIVE#  %01%A  PAK SENT %01%A FLAG RCVD %03%N/
                                        .EVEN
012654      045     101     102  RECID:: .ASCIZ  /%ABLOCK#  %04%A COMMAND %02%A EXPCTD %03%A SUCCESS %03%N/
                                        .EVEN
012746      103     101     116  OVRFLO: .ASCIZ  /CAN'T UPDATE ERROR OR STATISTIC:OVERFLOW PENDING/
                                        .EVEN
013030      045     101     040  RECID2: .ASCIZ  /%A RCDB WAS %0c%N/
                                        .EVEN
```

```
                                .SBTTL  WHCHDR / SEE WH'CH DRIVE IS ACTIVE

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

013052  000241                  WHCHDR:: CLC                    ;CLEAR CARRY

013054  105765  000060                 TSTB    DR(R5)         ;DR 0?
013060  001401                  BEQ     2$                     ;YES
013062  000261                  SEC                            ;NO

013064  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
                                ;--


013066                          CHKSUM:: PUSH   R3
013070                                   PUSH   R2
013072  042737  000001  003300           BIC    #BIT0,$SYSTAT    ;"CHECKSUM IS ODD" BIT
013100  032701  000001                    BIT    #BIT0,R1         ;AN ODD # OF BYTES?
013104  001403                           BEQ    1$               ;NO
013106  052737  000001  003300           BIS    #BIT0,$SYSTAT    ;YES

013114  006001                  1$:      ROR    R1               ;/2 FOR WORDS

013116  005003                  2$:      CLR    R3               ;PREP CHECKSUM WORD

013120  062003                  3$:      ADD    (R0)+,R3         ;FORM SUM
013122  005503                           ADC    R3               ;WITH CARRY
013124  005301                           DEC    R1               ;MORE WORDS?
013126  001374                           BNE    3$               ;YES

013130  032737  000001  003300           BIT    #BIT0,$SYSTAT    ;WAS IT ODD
013136  001405                           BEQ    4$               ;NO
013140  112002                           MOVB   (R0)+,R2         ;YES GET NEXT BYTE
013142  042702  177400                   BIC    #177400,R2       ;UN-SIGN EXTEND
013146  060203                           ADD    R2,R3            ;ADD IT IN
013150  005503                           ADC    R3               ;AND CARRY JUST IN CASE

013152  010301                  4$:      MOV    R3,R1            ;RETURN IT IN CORRECT PLACE
013154                                   POP    R2               ;RESTORE
013156                                   POP    R3
013160  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
                                ;--


013162                          CKCKSM:: PUSH   R1
013164                                   PUSH   R0              ;SAVE
013166  010401                          MOV    R4,R1           ;COPY BYTE COUNT TO CORRECT
013170  004737  013066                   CALL   CHKSUM          ;REGISTER FOR CHKSUM AND
                                                                ;FORM CHECKSUM

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

013174  122001                           CMPB   (R0)+,R1        ;LOWER ORDER CHECK
013176  001005                           BNE    2$              ;WRONG

013200  000301                           SWAB   R1              ;OK-PREP FOR

013202  122001                           CMPB   (R0)+,R1        ;HIGH ORDER CHECK
013204  001002                           BNE    2$              ;WRONG
013206  000241                           CLC                    ;OK-CLEAR SAILING

013210  000401                           BR     3$              ;EXIT

013212  000261                  2$:      SEC                    ;LET ERROR BE KNOWN


013214                          3$:      POP    R0
013216                                   POP    R1
013220  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
                                        ;--

013222  105037  013755          DOBRK:: CLRB    INITWD+1        ;CLEAR BYTE RECEIVE ADDR
013226  005037  013756                  CLR     BRKTO           ;CLEAR TIME OUT CONSTANT
013232  052775  000001  000026        1$:     BIS     #BIT0,@XMSR(R5) ;SET 'BREAK'
013240  012765  000001  000100                MOV     #RSSNIT,CMDSNT(R5)  ;SAY WE SENT 'INIT'
013246  052715  000020                  BIS     #BIT4,@R5       ;PAK SENT TYPE =COMMAND, SORT OF
013252  012704  000010                  MOV     #8.,R4          ;BREAK-IT'S-BACK COUNT-8
013256                                  BREAK                   ;SUPERVISOR TAKE FIVE
                                                                ;FOR ^C CHECK, ETC.
013260  105775  000026                  TSTB    @XMSR(R5)       ;READY?
013264  100410                          BMI     4$              ;YES
013266  005337  013756                  DEC     BRKTO           ;NO, TIME OUT?
013272  001371                          BNE     1$              ;NO
013274  012704  000056                  MOV     #TOSNDB,R4      ;YES, SET ERROR CODE
013300  004737  012046                  CALL    LOG             ;LOG IT
013304  000535                          BR      3$              ;EXIT
013306  113775  013752  000030  4$:     MOVB    BRKWD,@XMDB(R5) ;SEND NULL
013314  005037  013756                  CLR     BRKTO           ;RESET TIME OUT
013320  005304                          DEC     R4              ;MORE NULLS TO SEND?
013322  001355                          BNE     1$              ;YES
013324  005075  000026                  CLR     @XMSR(R5)       ;NO, CLEAR 'BREAK'
013330  017500  000024                  MOV     @RCDB(R5),R0    ;HEAVE 'GARBAGE' 1ST BYTE
013334                                  SETPRI  #PR100          ;SET TO INTERRUPT FO SURE
013342                                  SETVEC  TUVECT(R5),#RCVINT,#PR107 ;SET VECTO INFO
013370  062765  000004  000204          ADD     #4,TUVECT(R5)   ;AND INC TO SND VECTOR
013376                                  SETVEC  TUVECT(R5),#SNDINT,#PR107;AND SET IT
013424  162765  000004  000204          SUB     #4,TUVECT(R5)   ;RESET VECTOR ADDR.
013432  005037  013756                  CLR     BRKTO           ;RESET TIME OUT
013436  012704  013754                  MOV     #INITWD,R4      ;USE ADDR. FOR SNDBYT
013442  010437  013760                  MOV     R4,BRKPTR       ;AND SAVE FOR "WAIT"
013446  052775  000100  000026          BIS     #BIT6,@XMSR(R5) ;ENABLE INTER.
013454  004737  013716                  CALL    WAIT            ;AND ENTER LOOP
013460  005715                          TST     @R5             ;ABORTED FROM TIME OUT?
013462  100446                          BMI     3$              ;YES-EXIT

013464  005037  013756                  CLR     BRKTO           ;RESET TIME OUT
```

```
2677 013470  012704  013754                    MOV     #INITWD,R4      ;SEND SECOND INIT
2678 013474  010437  013760                    MOV     R4,BRKPTR       ;SAVE POINTER AGAIN
2679 013500  052775  000100  000026            BIS     #BIT6,@XMSR(R5) ;AND THEN ENABLE INT
2680 013506  004737  013716                    CALL    WAIT            ;AND WAIT
2681 013512  005715                            TST     @R5             ;IF ABORTED
2682 013514  100431                            BMI     3$              ;THEN EXIT
2683
2684 013516  012704  013755                    MOV     #INITWD+1,R4    ;WHERE RESPONSE WILL GO (ADDRESS)
2685 013522  010437  013760                    MOV     R4,BRKPTR       ;AND FOR 'WAIT'
2686 013526  052775  000100  000022            BIS     #BIT6,@RCSR(R5) ;ENABLE RECIEVE INT.
2687 013534  004737  013716                    CALL    WAIT            ;GET ANSWER
2688 013540  005715                            TST     @R5             ;ABORTED?
2689 013542  100416                            BMI     3$              ;YES.
2690
2691 013544  123727  013755  000020            CMPB    INITWD+1,#RSCONT  ;NO, IS IT 'CONTINUE'?
2692 013552  001003                            BNE     2$              ;NOPE-ERROR
2693
2694 013554  042715  040000                    BIC     #BIT14,@R5      ;SUCCESSFUL, CLEAR DOBREAK FLAG
2695 013560  000407                            BR      3$              ;EXIT
2696
2697 013562  113737  013755  003301  2$:       MOVB    INITWD+1,SYSTAT+1       ;SAVE BUM RESPONSE
2698 013570  012704  000032                    MOV     #CNINIT,R4              ;CAN'T INIT CODE
2699 013574  004737  012046                    CALL    LOG             ;LOG IT
2700                                                                    ;SCHEDULER WILL TRY AGAIN IF NOT ABORTED
2701
2702 013600  042775  000100  000026  3$:       BIC     #BIT6,@XMSR(R5) ;CLEAR INTERRUPTS
2703 013606  042775  000100  000022            BIC     #BIT6,@RCSR(R5) ; AND FOR RECIEVE
2704 013614                                    CLRVEC  TUVECT(R5)      ;RELEASE RECIEVE VECT.
2705 013622  062765  000004  000204            ADD     #4,TUVECT(R5)   ;AND GET SEND ADDR.
2706 013630                                    CLRVEC  TUVECT(R5)      ;AND RELEASE IT
2707 013636  162765  000004  000204            SUB     #4,TUVECT(R5)   ;RESTORE POINTER
2708 013644  000207                            RETURN                  ;RETURN
```

```
                                .SBTTL  INTERRUPT SERVICE ROUTINES AND TIMER

013646                          BGNSRV  SNDINT                   ;"SEND" INTERRUPT SERVICE:

013646  042775  000100  000026  SNDHND: BIC     #BIT6,@XMSP(R5) ;DISABLE INTERRUPT
013654  112475  000030                  MOVB    (R4)+,@XMDB(R5);OUTPUT BYTE
013660                          ENDSRV


013662                          BGNSRV  RCVINT                   ;"RCV" INTERRUPT SERVICE:

013662  042775  000100  000022  RCVHND: BIC     #BIT6,@RCSR(R5) ;DISABLE INTS
013670  017565  000024  000074          MOV     @RCDB(R5),DLV(R5)       ;SAVE BYTE
013676  116524  000074                  MOVB    DLV(R5),(R4)+   ;BYTE TO BUFFER
013702  005765  000074                  TST     DLV(R5)         ;ERROR?
013706  100402                          BMI     10$             ;YES
013710  005065  000074                  CLR     DLV(R5)         ;NO CLEAR ERROR
013714                          10$:
013714                          ENDSRV


013716  000240                  WAIT:   NOP                      ;WAIT  LOOP FOR
                                                                ;INTERRUPT SERVICING
013720  020437  013760                  CMP     R4,BRKPTR       ;IF=,THEN NO INTERRUPT
013724  001011                          BNE     1$              ;GOT ONE!
013726                                  BREAK                   ;SUPERVISOR BREAK
013730                                  BREAK                   ;KILL SOME TIME
013732  005337  013756                  DEC     BRKTO           ;TIME OUT?
013736  001367                          BNE     WAIT            ;NO...CONT.
013740  012704  000050                  MOV     #TORCVB,R4      ;YES LOAD ERROR #
013744  004737  012046                  CALL    LOG             ;LOG IT
013750  000207                  1$:     RETURN                  ;RETURN

013752  000000                  BRKWD:  .WORD   0               ;NULL
013754     004                  INITWD: .BYTE   RSINIT          ;INIT COMMAND
013755     000                          .BYTE   0               ;RSCONT IS EXPECTED HERE
013756  000000                  BRKTO:  .WORD   0               ;TIME OUT
013760  000000                  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
                                ;--

013762                          COMPAR:: PUSH    R0              ;COMPARE DATA IS DATA PACKET
013764                                   PUSH    R4              ;TO PATTERN WRITTEN
013766                                   PUSH    R1              ;USING BYTE COUNT IN PACKET
013770  005037  014140                   CLR     BDBYTS          ;CLEAR TOTAL WRONG
013774  016504  000104                   MOV     PKPTR(R5),R4    ;GET TOP OF PACKET
014000  005737  002212                   TST     CMPDAT          ;COMPARE SELECTED?
014004  001451                           BEQ     4$              ;NO-EXIT
014006  005204                           INC     R4              ;YES, LOCATE COUNT
014010  111401                           MOVB    @R4,R1          ;GET IT
014012  042701  177400                   BIC     #177400,R1      ;SIGN-UNEXTEND
                                                                 ;MUST TEST BYTE-WISE...
014016  005204                           INC     R4              ;-->FIRST DATA  BYTE
014020  126524  000072          1$:      CMPB    PATTEN(R5),(R4)+  ;DATA-WHAT WAS EXPECTED?
014024  001402                           BEQ     2$              ;YES
014026  005237  014140                   INC     BDBYTS          ;NO, INCREMENT TOTAL WRONG
014032  005301                  2$:      DEC     R1              ;MORE LEFT?
014034  001371                           BNE     1$              ;YES
014036  005737  014140                   TST     BDBYTS          ;ANY WRONG?
014042  001432                           BEQ     4$              ;NO
014044  011537  002074                   MOV     @R5,L$LUN       ;GET UNIT NUMBER
014050  042737  177770  002074           BIC     #177770,L$LUN   ;MASK IT OFF
014056                                   ERRSUFT 0.,MSBDA,ERRDES  ;YES-PRINT 'BAD DATA IN PACKET' ERROR
014066                                   PRINTB  #DESC,BDBYTS
014112  052715  000100                   BIS     #BIT6,@R5       ;LET 'EM KNOW UPSTAIRS-BAD DATA FLAG
014116  012737  000204  003330           MOV     #132.,PRNSIZ    ;SIZE IS ONE DATA PACK
014124  004737  014176                   CALL    PRNPAK          ;AND PRINT THE PACKET
014130                          4$:      POP     R1              ;RESTORE
014132                                   POP     R4
014134                                   POP     R0

014136  000207                           RETURN

014140  000000                  BDBYTS:  .WORD
014142     045     10     124   DESC:    .ASCIZ  /%ATOTAL BAD BYTES- %D3%A.%N/
                                .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
                                ;--

014176  000240                  PRNPAK:: NOP                    ;PRINTS 1 PACKET
                                                                ;PKPTR(R5)->TOP OF PACKET
                                                                ;PRNSIZ (PASSED)=BYTE COUNT
014200                                   PUSH    R0
014202                                   PUSH    R4
014204  105737  002210                 TSTB    PRBUF           ;PRINT PACKET SELECTED?
014210  001451                         BEQ     4$              ;NO
014212  016504  000104                 MOV     PKPTR(R5),R4    ;YES-GET TOP OF PACK
014216  012737  000020  014342  1$:    MOV     #16.,LNCNT      ;16 BYTES PER LINE
014224  112437  014344          2$:    MOVB    (R4)+,PRDAT     ;AVOID SIGN EXTEND
014230                                   PRINTF  #PRFORM,<B,PRDAT>  ;PRINT BYTE
014256  005337  003330                 DEC     PRNSIZ          ;ONE LESS
014262  001414                         BEQ     3$              ;NO MORE
014264  005337  014342                 DEC     LNCNT           ;NEW LINE?
014270  001355                         BNE     2$              ;NOT YET
014272                                   PRINTF  #CARLF          ;YES
014312  000741                         BR      1$              ;NEXT LINE
014314                          3$:    PRINTF  #CARLF          ;FINISH UP
014334                          4$:    POP     R4
014336                                   POP     R0
014340  000207                         RETURN                  ;RETURN

014342  000000                  LNCNT:  .WORD
014344  000000                  PRDAT:  .WORD
014346     045     117     063  PRFORM: .ASCIZ  /%03%A /
                                        .EVEN
014356     045     116     000  CARLF:  .ASCIZ  /%N/
                                        .EVEN

                                        ENDMOD
014362
```

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

014362                                  BGNMOD

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

014362                                  BGNRPT
014362                                  PUSH    R0
014364                                  PUSH    R1
014366                                  PUSH    R2
014370                                  PUSH    R3
014372                                  PUSH    R4
014374                                  PUSH    R5

014376                                  BREAK
014400  012737  003340  015010          MOV     #BLKTBL,RPTR    ;GET 1ST DEVICE BLOCK
014406                                  PRINTS  #STATHD         ;HEADER
014426                                  BREAK                   ;^C CHECK
014430                                  PRINTS  #STHD2          ;2ND HEADER
014450                          1$:     BREAK                   ;^C CHECK
014452  017705  000332                  MOV     @RPTR,R5        ;GET DEVICE BLOCK
014456  032715  004000                  BIT     #BIT11,@R5      ;UNIT NOT TESTED?
014462  001131                          BNE     2$              ;TRUE, DON'T PRINT STATISTICS
                                                                ;OK TO PRINT
014464  011537  015006                  MOV     @R5,RLUN        ;SAVE STATUS WORD
014470  042737  177770  015006          BIC     #177770,RLUN    ;MASK UNIT NUM.
014476  116501  000122                  MOVB    SOFTR(R5),R1    ;SOFTREAD
014502  042701  177400                  BIC     #177400,R1      ;SIGN-UNEXTEND
014506  116502  000124                  MOVB    SOFTW(R5),R2    ;SOFT WRITE
014512  042702  177400                  BIC     #177400,R2      ;
014516  116503  000136                  MOVB    HARDR(R5),R3    ;HARD READ
014522  042703  177400                  BIC     #177400,R3      ;
014526  116504  000140                  MOVB    HARDW(R5),R4    ;HARD WRITE
014532  042704  177400                  BIC     #177400,R4      ;
014536                                  PRINTS  #FMO,RLUN       ;SUMMARY/UNIT #
014562                                  PRINTS  #FM,#0,WRTN0(R5),RDN0(R5),<B,BDATA(R5)>,R1,R2,R3,R4
014634  116501  000123                  MOVB    SOFTR+1(R5),R1  ;SAME
014640  042701  177400                  BIC     #177400,R1      ;AS
014644  116502  000125                  MOVB    SOFTW+1(R5),R2  ;ABOVE
014650  042702  177400                  BIC     #177400,R2      ;THIS
014654  116503  000137                  MOVB    HARDR+1(R5),R3  ;TIME
014660  042703  177400                  BIC     #177400,R3      ;FOR
014664  116504  000141                  MOVB    HARDW+1(R5),R4  ;DRIVE
014670  042704  177400                  BIC     #177400,R4      ;ONE

014674                                  PRINTS  #FM,#1,WRTN1(R5),RDN1(R5),<B,BDATA+1(R5)>,R1,R2,R3,R4
014746  023727  015010  003356  2$:     CMP     RPTR,#LSTDEV    ;ALL UNITS DONE?
014754  103005                          BHIS    3$              ;YES
014756  062737  000002  015010          ADD     #2,RPTR         ;NO-DO

014764  000137  014450                  JMP     1$              ;MORE UNITS

014770                          3$:     POP     R5
```

```
2940 014772                                    POP     R4
2941 014774                                    POP     R3
2942 014776                                    POP     R2
2943 015000                                    POP     R1
2944 015002                                    POP     R0
2945 015004                                    ENDRPT
2946 015006  000000                    RLUN:   .WORD
2947 015010  000000                    RPTR:   .WORD
2948
2949 015012     045     116     045    STATHD: .ASCII  /%N%A      DR BLKS WR  BLKS RD  BDPAK  /
2950 015060     104     103     110            .ASCIZ  @DCHK/RD DCHK/WR  DCHK/RD DCHK/WR%N@
2951                                           .EVEN
2952 015124     045     101     125    FMO:    .ASCIZ  /%AUNIT %D1%N/
2953                                           .EVEN
2954
2955 015142     045     101     040    FM:     .ASCII  /%A       %D1%A %D5%A.   %D5%A.    %D3%A.    /
2956 015216     045     104     063            .ASCIZ  /%D3%A.   %D3%A.     %D3%A.    %D3%A.%N/
2957                                           .EVEN
2958 015266     045     101     040    STHD2:  .ASCII  /%A                                       /
2959 015333     122     105     103            .ASCIZ  /RECOV   RECOV   UNRECOV UNRECOV%N/
2960                                           .EVEN
2961 015376                                    ENDMOD
```

```
                                        .SBTTL  INITIALIZE SECTION

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

015376                                          BGNINIT

015376  000240                  INIT:   NOP                     ;
015400  105037  016150                  CLRB    STRT            ;FOR STATS CLEAR
015404                                  READEF  #EF.START       ;START COMMAND?
015412                                  BNCOMPLETE INIT2        ;NO
015414  005237  016150                  INC     STRT            ;YES. SET START FLAG
015420  012737  003340  003304  INIT2:  MOV     #BLKTBL,DEVPTR  ;SET ALL UNITS ABORTED:
015426  005004                          CLR     R4              ;UNIT NUMBER
015430  017705  165650          1$:     MOV     @DEVPTR,R5      ;GET POINTER
015434  010415                          MOV     R4,@R5          ;INSERT UNIT #
015436  052715  120000                  BIS     #BIT15!BIT13,@R5  ;SET ABORTED, HALTED
015442  052715  004000                  BIS     #BIT11,@R5      ;SET UNIT NOT TESTED
015446  006304                          ASL     R4              ;*2 FOR LOOK-UP
015450  016465  024346  000102          MOV     BUFTBL(R4),RCVBUF(R5)  ;SETUP POINTER TO UNIT'S BUFFER
015456  006204                          ASR     R4              ;CORRECT BACK TO UNIT #
015460  023727  003304  003356          CMP     DEVPTR,#LSTDEV  ;LAST DEVICE DONE?
015466  103005                          BHIS    CHECK           ;YES
015470  062737  000002  003304          ADD     #2,DEVPTR       ;NO-GET
015476  005204                          INC     R4              ;NEXT DEVICE AND
015500  000753                          BR      1$              ;SERVICE

015502  022737  000010  002012  CHECK:  CMP     #8.,L$UNIT      ;MAKE SURE NOT
015510  103005                          BHIS    GETHRD          ;TOO MANY UNITS
015512                                  ERRSF   101.,TOMANY             ;TOMANY-REQUEST ^C
015522                                  DOCLN                   ;EXIT

015524  012737  003340  003304  GETHRD: MOV     #BLKTBL,DEVPTR  ;INIT TABLE POINTER
015532  005004                          CLR     R4              ;CLEAR DEVICE COUNTER
015534  017705  165544          1$:     MOV     @DEVPTR,R5      ;GET STATUS WORD
015540  010437  002074                  MOV     R4,L$LUN        ;UNIT NUM. IN CASE ERROR
015544                                  GPHARD  R4,R2           ;GET HARD INFO
015552                                  BNCOMPLETE 3$
015554  042715  004000                  BIC     #BIT11,@R5      ;UNIT IS TESTED!
015560  012203                          MOV     (R2)+,R3        ;R3=CSR
015562  012265  000204                  MOV     (R2)+,TUVECT(R5)  ;GET VECTOR ADDRESS
015566  112265  000061                  MOVB    (R2)+,DR+1(R5)  ;SAVE UNIT SUMMARY
015572  005202                          INC     R2              ;GET TO WORD BOUND
015574  012237  016152                  MOV     (R2)+,PDTFLG    ;AND GET PDT FLAG
015600  052715  040000                  BIS     #BIT14,@R5      ;SET SEND BREAK FLAG
015604  032765  000400  000060          BIT     #BIT8,DR(R5)    ;DRIVE 0?
015612  001011                          BNE     13$             ;YES
015614  032765  001000  000060          BIT     #BIT9,DR(R5)    ;DRIVE 1?
015622  001005                          BNE     13$             ;OK
015624                                  ERRSF   102.,NODRVS             ;NEITHER?.
015634                                  DOCLN                   ;EXIT

015636  105737  016150          13$:    TSTB    STRT            ;START COMMAND?
015642  001412                          BEQ     14$             ;NO, DONT CLEAR
                                                                ;YES-CLEAR STATS
```

```
3021 015644  012702  000202                  MOV     #BLKEND,R2        ;R2-->END OF STATS
3022 015650  012701  000110                  MOV     #WRTNO,R1         ;FORM ADDRESS OF START:
3023 015654  060501                          ADD     R5,R1             ;R1-->START OF STATS.
3024 015656  162702  000110                  SUB     #WRTNO,R2         ;FORM # TO CLEAR
3025
3026 015662  105021                  2$:     CLRB    (R1)+             ;CLEAR 'EM
3027 015664  005302                          DEC     R2                ;MORE?
3028 015666  001375                          BNE     2$                ;YES
3029 015670  042715  120000          14$:    BIC     #BIT15!BIT13,@R5  ;SET NOT ABORTED NOT HALTED
3030 015674  010365  000022                  MOV     R3,RCSR(R5)       ;GET DEVICE REGISTERS:
3031 015700  062703  000002                  ADD     #2,R3
3032 015704  010365  000024                  MOV     R3,RCDB(R5)
3033 015710  062703  000002                  ADD     #2,R3
3034 015714  010365  000026                  MOV     R3,XMSR(R5)
3035 015720  062703  000002                  ADD     #2,R3
3036 015724  105737  016152                  TSTB    PDTFLG            ;UNIT A PDT?
3037 015730  001402                          BEQ     4$                ;NO
3038 015732  162703  000004                  SUB     #4,R3             ;YES...RCDB=XMDB
3039 015736  010365  000030          4$:     MOV     R3,XMDB(R5)
3040 015742  005065  000072                  CLR     PATTEN(R5)        ;ZERO DATA PATTERN
3041 015746  005065  000002                  CLR     RETRY(R5)         ;NO RETRIES
3042 015752  005065  000064                  CLR     REC(R5)           ;NO RECORD
3043 015756  005065  000076                  CLR     SUCCS(R5)         ;NO SUCCESS
3044 015762  005065  000074                  CLR     DLV(R5)           ;NO DLV ERROR
3045 015766  005037  003332                  CLR     ALLGON            ;OK TO PRINT STATISTICS
3046 015772  062737  000002  003304  3$:     ADD     #2,DEVPTR         ;-->NEXT DEVICE
3047 016000  005204                          INC     R4                ;INCREMENT UNIT NUMBER
3048 016002  020437  002012                  CMP     R4,L$UNIT         ;MORE UNITS?
3049 016006  001252                          BNE     1$                ;YES, GP HARD THE NEXT
3050
3051 016010  005037  003300                  CLR     SYSTAT            ;SYSTEM STATUS WORD
3052 016014                                  RFLAGS  FLGLOC            ;GET USER FLAGS
3053 016022  005037  003324          5$:     CLR     BLKER             ;NO ERROR
3054 016026  013737  002204  003302  SETLEN: MOV     LENGTH,TAPLEN     ;GET # OF RECOR.S
3055 016034  006237  003302                  ASR     TAPLEN            ;GET # BLOCKS PER TRACK
3056 016040  012737  000200  003326          MOV     #200,SECREC       ;PRESET SECOND START AT 200
3057 016046  022737  000200  003302          CMP     #200,TAPLEN       ;# BLKS > 128.?
3058 016054  101003                          BHI     3$                ;NO-SWITCH TRACKS 2ND PASS
3059 016056  012737  000400  003326          MOV     #400,SECREC       ;YES-START AT 400
3069
3081
3082 016064                          3$:     ENDINIT
3083
3084
3085 016066     124     117     117  TOMANY: .ASCIZ  /TOO MANY UNITS MAX.=8 /
3086                                         .EV'
3087 016116     123     105     114  NODRVS: . ..IZ  /SELECT AT LEAST 1 DRIVE /
3088                                         .EVEN
3089 016150  000000                  STRT::  .WORD
3090 016152  000000                  PDTFLG::.WORD             ;TU58 IS IN PDT
3091 016154  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.
                                        ;--

016156                                  BGNAUTO
016156  000240                                NOP             ;AUTO DROP ROUTINE
016160                                      SETVEC  #4,#TRPHND,#PR107       ;GET BUS TRAP VEC.
016206  012737  003340  016264              MOV     #BLKTBL,TRPPTR  ;GET TOP OF DATA BLOCK TABLE
016214  017705  000044          1$:         MOV     @TRPPTR,R5      ;GET DATA BLOCK
016220  032715  104000                      BIT     #BIT15!BIT11,@R5  ;NOT TESTED OR ABORTED?
016224  100403                              BMI     2$              ;YES
016226  005775  000022                      TST     @RCSR(R5)       ;NO-VALID ADDRESS?
016232  000240                              NOP                     ;YES...(TRAP IF NOT)
016234  023727  016264  003'56  2$:         CMP     TRPPTR,#LSTDEV ;MORE TO TRY?
016242  103004                              BHIS    3$              ;NO
016244  062737  000002  016264              ADD     #2,TRPPTR       ;ON TO NEXT
016252  000760                              BR      1$              ;GET IT
016254                                  3$: CLRVEC  #4              ;RESTORE
016262                                  ENDAUTO
016264  000000                          TRPPTR: .WORD


                                        ;ILLEGAL ADDRESS TRAP HANDLER:

016266                                  TRPHND: PRINTF  #MSAUTO         ;SAY "AUTO DROPPED"
016306  011500                              MOV     @R5,R0          ;GET UNIT #
016310  042700  177770                      BIC     #177770,R0      ;MASK IT OFF
016314                                      DODU    R0              ;DROP HIM
016316  000002                              RTI
016320    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.
                                ;--

016340                                  BGNCLN
016340  005737  003332                  TST     ALLGON          ;ENTRANCE FROM ALL-UNITS-ABORTED?
016344  001004                          BNE     1$              ;YES-EXIT
016346  005737  002206                  TST     STAEOP          ;NO-STATS AT EOP?
016352  001401                          BEQ     1$              ;NO
016354                                  DORPT                   ;YES

016356                          1$:     ENDCLN
```

```
                                .SBTTL  DROP UNIT SECTION

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

016360                                  BGNDU
                                                        ;R0=UNIT NUMBER
016360                                  PUSH    R0      ;SAVE IT
016362                                  PUSH     R5     ;SAVE PRESENT UNIT POINTER
016364  004737  016424                  CALL     GETR5  ;GET POINTER TO UNIT
016370  052715  120000                  BIS     #BIT15!BIT13,@R5  ;SET ABORTED, HALTED
016374                                  POP     R5      ;RESTORE PRESENT UNIT POINTER
016376                                  POP     R0      ;RETRIEVE UNIT NUMBER
016400                                  PRINTF  #ABOMSG,R0

                                        ENDDU
016422
016424  012737  003340  016454  GETR5: MOV     #BLKTBL,PTR     ;-->UNIT 0
016432  017705  000016          1$:     MOV     @PTR,R5         ;GET STATUS WORD
016436  005300                          DEC     R0              ;CORRECT UNIT?
016440  100404                          BMI     2$              ;YES
016442  062737  000002  016454          ADD     #2,PTR          ;NO,-->NEXT
016450  000770                          BR      1$              ;CONTINUE
016452  000207                  2$:     RETURN
016454  000000                  PTR:    .WORD

016456      045     101     104 ABOMSG: .ASCIZ  /%ADROPPED UNIT %D1%N/
32 1                                    .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.
                                ;--

016504                                  BGNAU

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




016504                                  ENDAU

```

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

                                        BGNMOD
                                .NLIST  ME,BEX

                                        BGNTST
                                        TSTID   #TST1
016506  012737  016552  003220                        MOV     #TST1,TSTTOP    ;SAVE ADDR OF TEST
016514  004737  005754                                CALL    SETUP           ;INIT UNITS TSTPC
016520  004737  005602                                CALL    SETDR           ;GET 1ST DRVS.
016524  004737  006022                                CALL    RUN             ;DO TEST
016530  004737  005500                                CALL    SWAPDR          ;GET NEXT DRVS.
016534  103004                                        BCC     64$             ;BR NO 2ND DRVS
016536  004737  005754                                CALL    SETUP           ;REINIT UNITS TSTPC
016542  004737  006022                                CALL    RUN             ;REPEAT TEST
016546                                          64$:                          ;DONE
                                        EXIT TST

                                TST1:   TUSELF
                                        INC     DONE
                                        RETURN


                                        ENDTST
```

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

                                        BGNTST
                                        TSTID   #TST2
016702  012737  016746  003320                        MOV     #TST2,TSTTOP    ;SAVE ADDR OF TEST
016710  004737  005754                                CALL    SETUP           ;INIT UNITS TSTPC
016714  004737  005602                                CALL    SETDR           ;GET 1ST DRVS.
016720  004737  006022                                CALL    RUN             ;DO TEST
016724  004737  005500                                CALL    SWAPDR          ;GET NEXT DRVS.
016730  103004                                        BCC     64$             ;BR NO 2ND DRVS
016732  004737  005754                                CALL    SETUP           ;REINIT UNITS TSTPC
016736  004737  006022                                CALL    RUN             ;REPEAT TEST
016742                                          64$:                          ;DONE
                                        EXIT TST

                                TST2:   CLR     R4              ;R4=INDEX INTO RECORD TABLE
016746  005004
016750  016465  017130  000064    1$:   MOV     RECDAT(R4),REC(R5)  ;GET THE RECORD

016756                                  TUSEEK  REC(R5),DR(R5)  ;SEEK IT

017106  062704  000002                ADD     #2,R4           ;POINT TO NEXT RECORD
017112  026427  017130  177777        CMP     RECDAT(R4),#-1. ;LAST ONE DONE?
017120  001313                        BNE     1$              ;NO-LOOP
017122  005237  003314                INC     DONE            ;YES-SET DONE FLAG
017126  000207                        RETURN

017130  000000                RECDAT: 0.      ;BOT
017132  000200                        200     ;BOT OTHER TRACK
017134  000177                        177     ;EOT
017136  000377                        377     ;EOT OTHER TRACK
017140  000400                        400     ;BOT AGAIN
017142  177777                        -1.
017144                                  ENDTST
```

```
                                .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.

017146                                  BGNTST
017146                                  TSTID   #TST3
017146  012737  017212  003320                        MOV     #TST3,TSTTOP    ;SAVE ADDR OF TEST
017154  004737  005754                                CALL    SETUP           ;INIT UNITS TSTPC
017160  004737  005602                                CALL    SETDR           ;GET 1ST DRVS.
017164  004737  006022                                CALL    RUN             ;DO TEST
017170  004737  005500                                CALL    SWAPDR          ;GET NEXT DRVS.
017174  103004                                        BCC     64$             ;BR NO 2ND DRVS
017176  004737  005754                                CALL    SETUP           ;REINIT UNITS TSTPC
017202  004737  006022                                CALL    RUN             ;REPEAT TEST
017206                                          64$:                            ;DONE
017206                                  EXIT TST

017212  012765  000100  000066  TST3:   MOV     #100,TMP(R5)    ;INIT TO HALF OF REMAINING
017220  005004                          CLR     R4              ;FOR INDEX INTO DATA TABLE
017222  005065  000064                  CLR     REC(R5)         ;START AT RECORD 0
017226  016465  020522  000072  1$:     MOV     TST3PT(R4),PATTEN(R5)   ;GET DATA
017234                                  TUWRIT  PATTEN(R5),REC(R5),#512.,DR(R5),#0
020024                                  TUREAD  REC(R5),#512.,DR(R5),#0
020424  062704  000002                  ADD     #2,R4           ;POINT TO NEXT DATA
020430  005764  020522                  TST     TST3PT(R4)      ;END?
020434  001402                          BEQ     2$              ;YES
020436  000137  017226                  JMP     1$              ;NO-WRITE, READ NEW DATA
020442  005004                  2$:     CLR     R4              ;POINT TO FIRST DATA
020444  062765  000200  000064          ADD     #200,REC(R5)    ;BUT NOW USE ADJACENT RECORD
020452  032765  001000  000064          BIT     #1000,REC(R5)   ;ALL ADJACENT RECORDS DONE?
020460  001002                          BNE     3$              ;YES
020462  000137  017226                  JMP     1$              ;NO-WRITE, READ AT NEW RECORD
020466  162765  001000  000064  3$:     SUB     #1000,REC(R5)   ;RESTORE TO NEXT RECORD
020474  066565  000066  000064          ADD     TMP(R5),REC(R5) ;HALF INTO REST OF TAPE
020502  006265  000066                  ASR     TMP(R5)         ;HALF OF HALF FOR NEXT TIME
020506  103402                          BCS     4$              ;DONE?
020510  000137  017226                  JMP     1$              ;NO
020514  005237  003314          4$:     INC     DONE            ;YES-SET FLAG
020520  000207                          RETURN
020522  000000                  TST3PT: .WORD   000000
020524  125252                          .WORD   125252
020526  177777                          .WORD   177777
020530  052525                          .WORD   052525
020532  000000                          .WORD   000000

020534                                  ENDTST
```

```
3391
3392                                    .SBTTL  TEST 4 / WRITE SELECTED NUMBER OF BLOCKS
3393
3394 020536                                    BGNTST
3395 020536                                    TSTID   #TST4
     020536  012737  020602  003320                            MOV     #TST4,TSTTOP    ;SAVE ADDR OF TEST
     020544  004737  005754                                    CALL    SETUP           ;INIT UNITS TSTPC
     020550  004737  005602                                    CALL    SETDR           ;GET TST DRVS.
     020554  004737  006022                                    CALL    RUN             ;DO TEST
     020560  004737  005500                                    CALL    SWAPDR          ;GET NEXT DRVS.
     020564  103004                                            BCC     64$             ;BR NO 2ND DRVS
     020566  004737  005754                                    CALL    SETUP           ;REINIT UNITS TSTPC
     020572  004737  006022                                    CALL    RUN             ;REPEAT TEST
     020576                                         64$:                               ;DONE
3396 020576                                    EXIT TST .
3397
3398
3399 020602  005065  000064            TST4:   CLR     REC(R5)         ;START AT REC 0
3400 020606  013765  003302  000066            MOV     TAPLEN,TMP(R5)  ;GET THE # OF BLOCKS PER TRACK
3401 020614  005065  000062            1$:     CLR     TRK(R5)         ;TRK(R5)=1ST OR 2ND PASS COUNTER
3402 020620  016565  000064  000072            MOV     REC(R5),PATTEN(R5)  ;USE RECORD NO. FOR DATA
3403 020626  005737  002214                    TST     DRVCHK                  ;ADD DR #?
3404 020632  001403                            BEQ     10$                     ;NO
3405 020634  066565  000060  000072            ADD     DR(R5),PATTEN(R5)       ;YES, ADD DRIVE ID
3406 020642                            10$:    TUWRIT  PATTEN(R5),REC(R5),#512.,DR(R5),#0
3407 021432  005365  000066                    DEC     TMP(R5)         ;DO ALL RECORDS FOR THIS TRACK?
3408 021436  001404                            BEQ     2$              ;YES-GET OTHER TRACK
3409 021440  005265  000064                    INC     REC(R5)         ;NO-ONTO NEXT RECORD
3410 021444  000137  020620                    JMP     1$              ;EXECUTE THE WRITE
3411 021450  005765  000062            2$:     TST     TRK(R5)         ;DONE 2 TRACKS?
3412 021454  001012                            BNE     TST4EX          ;YES-EXIT
3413 021456  005265  000062                    INC     TRK(R5)         ;NO-SET FLAG FOR NEXT PASS
3414 021462  013765  003326  000064            MOV     SECREC,REC(R5)  ;GET NEW STARTING BLOCK #
3415 021470  013765  003302  000066            MOV     TAPLEN,TMP(R5)  ;RESET # OF BLOCKS
3416 021476  000137  020620                    JMP     1$              ;AND EXECUTE
3417 021502  005237  003314            TST4EX: INC     DONE            ;DONE
3418 021506  000207                            RETURN                  ;RETURN
3419
3420 021510                                    ENDTST
```

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

                                        BGNTST
                                        TSTID   #TST5
021512  012737  021556  003320                        MOV     #TST5,TSTTOP    ;SAVE ADDR OF TEST
021520  004737  005754                                CALL    SETUP           ;INIT UNITS TSTPC
021524  004737  005602                                CALL    SETDR           ;GET TST DRVS.
021530  004737  006022                                CALL    RUN             ;DO TEST
021534  004737  005500                                CALL    SWAPDR          ;GET NEXT DRVS.
021540  103004                                        BCC     64$             ;BR NO 2ND DRVS
021542  004737  005754                                CALL    SETUP           ;REINIT UNITS TSTPC
021546  004737  006022                                CALL    RUN             ;REPEAT TEST
021552                                          64$:                          ;D .E
                                        EXIT TST

021556  005065  000064          TST5:   CLR     REC(R5)         ;START AT REC 0
021562  013765  003302  000066          MOV     TAPLEN,TMP(R5)  ;GET THE # OF BLOCKS PER TRACK
021570  005065  000062                  CLR     TRK(R5)         ;TRK(R5)=1ST OR 2ND PASS
021574  016565  000064  000072  1$:     MOV     REC(R5),PATTEN(R5)  ;USE RECORD NO. AS DATA
021602  005737  002214                  TST     DRVCHK          ;ADD DR #?
021606  001403                          BEQ     10$             ;NO
021610  066565  000060  000072          ADD     DR(R5),PATTEN(R5)  ;ADD IN DRIVE ID
021616                          10$:    TUREAD  REC(R5),#512.,DR(R5),#0
022216  005365  000066                  DEC     TMP(R5)         ;DO ALL RECORDS THIS TRACK?
022222  001404                          BEQ     2$              ;YES-GET OTHER TRACK
022224  005265  000064                  INC     REC(R5)         ;NO-NEXT RECORD
022230  000137  021574                  JMP     1$              ;EXECUTE THE READ
022234  005765  000062          2$:     TST     TRK(R5)         ;DONE 2 TRACKS?
022240  001012                          BNE     TST5E.          ;YES-EXIT
022242  005265  000062                  INC     TRK(R5)         ;NO-SET FLAG FOR NEXT PASS
022246  013765  003326  000064          MOV     SECREC,REC(R5)  ;GET NEW STARTING BLOCK #
022254  013765  003302  000066          MOV     TAPLEN,TMP(R5)  ;RESET # OF BLOCKS
022262  000137  021574                  JMP     1$              ;AND EXECUTE
022266  005237  003314          TST5EX: INC     DONE            ;DONE
022272  000207                          RETURN                  ;RETURN

022274                                  ENDTST
```

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

022276                                  BGNTST
022276                                  TSTID   #TST6
022276  012737  022342  003320                                  MOV     #TST6,TSTTOP    ;SAVE ADDR OF TEST
022304  004737  005754                                          CALL    SETUP           ;INIT UNITS TSTPC
022310  004737  005602                                          CALL    SETDR           ;GET 1ST DRVS.
022314  004737  006022                                          CALL    RUN             ;DO TEST
022320  004737  005500                                          CALL    SWAPDR          ;GET NEXT DRVS.
022324  103004                                                  BCC     64$             ;BR NO 2ND DRVS
022326  004737  005754                                          CALL    SETUP           ;REINIT UNITS TSTPC
022332  004737  006022                                          CALL    RUN             ;REPEAT TEST
022336                                                  64$:                            ;DONE
022336                                  EXIT TST

022342  005065  000064          TST6:   CLR     REC(R5)         ;START AT REC 0
022346  013765  003302  000066          MOV     TAPLEN,TMP(R5)  ;GET THE # OF BLOCKS PER TRACK
022354  005065  000062                  CLR     TRK(R5)         ;TRK(R5)=1ST OR 2ND PASS
022360  016565  000064  000072  1$:     MOV     REC(R5),PATTEN(R5)  ;USE RECORD NO. FOR DATA
022366  005737  002214                  TST     DRVCHK          ;ADD DR #?
022372  001403                          BEQ     10$             ;NO
022374  066565  000060  000072          ADD     DR(R5),PATTEN(R5)       ;ADD DRIVE ID
022402                          10$:    TUWRIT  PATTEN(R5),REC(R5),#512.,DR(R5),#1
023172  005365  000066                  DEC     TMP(R5)         ;DO ALL RECORDS FOR THIS TRACK?
023176  001404                          BEQ     2$              ;YES-GET OTHER TRACK
023200  005265  000064                  INC     REC(R5)         ;NO-NEXT RECORD
023204  000137  022360                  JMP     1$              ;EXECUTE THE WRITE
023210  005765  000062          2$:     TST     TRK(R5)         ;DONE 2 TRACKS?
023214  001012                          BNE     TST6EX          ;YES-EXIT
023216  005265  000062                  INC     TRK(R5)         ;NO-SET FLAG FOR NEXT PASS
023222  013765  003326  000064          MOV     SECREC,REC(R5)  ;GET NEW STARTING BLOCK #
023230  013765  003302  000066          MOV     TAPLEN,TMP(R5)  ;RESET # OF BLOCKS
023236  000137  022360                  JMP     1$              ;AND EXECUTE
023242  005237  003314          TST6EX: INC     DONE            ;DONE
023246  000207                          RETURN                  ;RETURN

023250                                  END.T
```

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

                                        BGNTST
                                        TSTID   #TST7
023252  012737  023314  003320                         MOV     #TST7,TSTTOP    ;SAVE ADDR OF TEST
023260  004737  005754                                 CALL    SETUP           ;INIT UNITS TSTPC
023264  004737  005602                                 CALL    SETDR           ;GET 1ST DRVS.
023270  004737  006022                                 CALL    RUN             ;DO TEST
023274  004737  005500                                 CALL    SWAPDR          ;GET NEXT DRVS.
023300  103004                                         BCC     64$             ;BR NO 2ND DRVS
023302  004737  005754                                 CALL    SETUP           ;REINIT UNITS TSTPC
023306  004737  006022                                 CALL    RUN             ;REPEAT TEST
023312                                          64$:                            ;DONE
                                        EXIT TST

023316  005065  000064          TST7:   CLR     REC(R5)         ;START AT REC 0
023322  013765  003302  000066          MOV     TAPLEN,TMP(R5)  ;GET THE # OF BLOCKS PER TRACK
023330  005065  000062                  CLR     TRK(R5)         ;TRK(R5)=1ST OR 2ND PASS
023334  016565  000064  000072  1$:     MOV     REC(R5),PATTEN(R5)  ;USE RECORD NO. FOR DATA
023342  005737  002214                  TST     DRVCHK          ;ADD DR #?
023346  001403                          BEQ      10$            ;NO
023350  066565  000060  000072          ADD     DR(R5),PATTEN(R5)       ;ADD DRIVE ID
023356                          10$:    TUREAD  REC(R5),#512.,DR(R5),#1
023756  005365  000066                  DEC     TMP(R5)         ;DO ALL RECORDS THIS TRACK?
023762  001404                          BEQ     2$              ;YES-GET OTHER TRACK
023764  005265  000064                  INC     REC(R5)         ;NO-NEXT RECORD
023770  000137  023334                  JMP     1$              ;EXECUTE THE READ
023774  005765  000062          2$:     TST     TRK(R5)         ;DONE 2 TRACKS?
024000  001012                          BNE     TST7EX          ;YES-EXIT
024002  005265  000062                  INC     TRK(R5)         ;NO-SET FLAG FOR NEXT PASS
024006  013765  003326  000064          MOV     SECREC,REC(R5)  ;GET NEW STARTING BLOCK #
024014  013765  003302  000066          MOV     TAPLEN,TMP(R5)  ;RESET # OF BLOCKS
024022  000137  023334                  JMP     1$              ;AND EXECUTE
024026  005237  003314          TST7EX: INC     DONE            ;DONE
024032  000207                          RETURN                  ;RETURN

                                        ENDTST
024034
```

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

```
                                .SBTTL  I/O BUFFER AREAS:

                                ;WHO-GETS-WHAT-SPACE TABLE

024346  025426                BUFTBL: .WORD   BUF0
024350  026464                        .WORD   BUF1
024352  027522                        .WORD   BUF2
024354  030560                        .WORD   BUF3
024356  031616                        .WORD   BUF4
024360  032654                        .WORD   BUF5
024362  033712                        .WORD   BUF6
024364  034750                        .WORD   BUF7


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

024366      023                             .BYTE RSXOFF
024367      023                             .BYTE RSXOFF    ;SEND XOFF BEFORE EVERY PACKET

024370                                      TRBUF:  .BLKB RCBFSZ
                                ;-------------------------------------------------------------


025426                          BUF0:   .BLKB   RCBFSZ
026464                          BUF1:   .BLKB   RCBFSZ
027522                          BUF2:   .BLKB   RCBFSZ
030560                          BUF3:   .BLKB   RCBFSZ
031616                          BUF4:   .BLKB   RCBFSZ
032654                          BUF5:   .BLKB   RCBFSZ
033712                          BUF6:   .BLKB   RCBFSZ
034750                          BUF7:   .BLKB   RCBFSZ

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

036006                                  ENDMOD
```

```
                                .TITLE PARAMETER CODING

                                .SBTTL  HARDWARE PARAMETER CODING SECTION

036006                                  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.
                                ;--

036006                                  BGNHRD

036010                                  GPRMA   MSG1,0,0,160000,177777,YES
036020                                  GPRMA   MSG1B,2,0,0,776,YES
036030                                  GPRML   MSG1C,6,1,YES
036036                                  GPRML   MSG2,4,1,YES
036044                                  GPRML   MSG3,4,2,YES

036052                                  ENDHRD

036052  124  125  065  MSG1:    .ASCIZ  /TU58 CSR/
036063  126  105  103  MSG1B:   .ASCIZ  /VECTOR ADDR./
036100  120  104  124  MSG1C:   .ASCIZ  /PDT (PARALLEL) INTERFACE/
036131  124  105  123  MSG2:    .ASCIZ  /TEST DRIVE 0/
036146  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.
                                ;--

036164                                  BGNSFT

036166                                  GPRMD   MSG4,0,D,1777,8.,512.,YES
036200                                  GPRML   MSG4B,10,1,YES
036206                                  GPRML   MSG5,2,1,YES
036214                                  GPRML   MSG6,6,1,YES
036222                                  GPRML   MSG7,4,1,YES

036230                                  GPRMD   MSG8,10.,D,377,1,254.,YES

036242                          SFTOUT: ENDSFT

036242  116     125     115     MSG4:   .ASCIZ  'NUMBER OF BLOCKS:TEST 4-7 (8 TO 512)'
036307  101     104     104     MSG4B:  .ASCIZ  /ADD DR # TO DATA PATTERN:TEST 4-7/
036351  123     124     101     MSG5:   .ASCIZ  /STATISTICS PRINTED AT EOP/
036403  103     117     115     MSG6:   .ASCIZ  /COMPARE DATA ON READ/
036430  120     122     111     MSG7:   .ASCIZ  /PRINT PACKET ON ERROR/
036456  043     040     105     MSG8:   .ASCIZ  /# ERRORS = DVC FATAL IF 'EVL'SET/
                                        .EVEN
```

```
3704          000006                          .REPT   14.             ;LASTAD CORRECTION
3705                                          .WORD
3706                                          .ENDR
3713 036554                                   LASTAD
     036560                         L$LAST::
3714 036560                                   ENDMOD
3715
3716 036560                                   BGNSETUP
3717 036560                                   BGNPTAB
3718 036564  176500                           176500
3719 036566  000300                           300
3720 036570  000003                           3
3721 036572  000000                           0
3722 036574                                   ENDPTAB
3723 036574                                   ENDSETUP
3724          000001                          .END
```

PARAMETER CODING MACRO M1110 02-AUG-79 15:53 PAGE 136-1 SEQ 0091
SYMBOL TABLE

```
ABNDX = 000004 G    CHKANS  010112 G    C$MSG = 000023      EVL   = 000004 G    G$RADA= 00C140
ABO     G12360      CHKEND  010536 G    C$OPEN= 000034      EVLTHR  002216      G$RADB= 000000
ABOMSG  016456      CHKERR  010624       C$PNTB= 000014      EXOFF   007213      G$RADD= C00040
ABONM   006336      CHKPKS  010202 G    C$PNTF= 000017      EXON    007212      G$RADL= 000120
ADR   = 000020 G    CHKPTR  010200       C$PNTS= 000016      E$END = 002100      G$RADO= 000020
ALLGON  003332 G    CHKREE  010676       C$PNTX= 000015      E$LOAD= 000035      G$XFER= 000004
ASSEMB= 000010      CHKRET  011242       C$QIO = 000377      FLGLOC  016154 G    G$YES = 000010
BDATA = 000134 G    CHKSUC  011536 G    C$RDBU= 000007      FM      015142      HARDR = 000136 G
BDBYTS  014140      CHKSUM  013066 G    C$REFG= 000047      FMO     0151?4      HARDW = 000140 G
BDCHK = 000022 G    CHKB    010132       C$RESE= 000033      FTLNM   003312      HELP  = 000000
BDCOM = 000014 G    CKCKSM  01 162 G    C$REVI= 000003      F$AU  = 000015      HOE   = 100000 G
BIT0  = 000001 G    CLRALL  C05660 G    C$RFLA= 000021      F$AUTO= 000020      HRD     012340
BIT00 = 000001 G    CLRBUF  005720 G    C$RPT = 000025      F$BGN = 000040      HRDRD = 000016 G
BIT01 = 000002 G    CLRPTR  005752       C$SEFG= 000046      F$CLEA= 000007      HRDWR = 000020 G
BIT02 = 000004 G    CMDSNT= 000100 G    C$SPRI= 000041      F$DU  = 000016      HRD1    011164
BIT03 = 000010 G    CMNDER= 000040 G    C$SVEC= 000037      F$END = 0.0041      IBE   = 010000 G
BIT04 = 000020 G    CMPDAT  002212       C$TPRI= 000013      F$HARD= 000004      IDPTR   003316 G
BIT05 = 000040 G    CNINIT= 000032 G    DESC    014142      F$HW  = 000013      IDU   = 000040 G
BIT06 = 000100 G    COMPAR  013762 G    DEVPTR  003304 G    F$INIT= 000006      IER   = 020000 G
BIT07 = 000200 G    CSNRDY  003334 G    DEV0    003360      F$JMP = 000050      INIT    015376
BIT08 = 000400 G    CSRCVB  003336 G    DEV1    003570      F$MOD = 000000      INITWD  013754
BIT09 = 001000 G    C$AU  = 000052       DEV2    004000      F$MSG = 000011      INIT2   015420
BIT1  = 000002 G    C$AUTO= 000061       DEV3    004210      F$PROT= 000021      ISR   = 000100 G
BIT10 = 002000 G    C$BRK = 000022       DEV4    004420      F$PWR = 000017      IXE   = 004000 G
BIT11 = 004000 G    C$BSEG= 000004       DEV5    004630      F$RPT = 000012      I$AU  = 000041
BIT12 = 010000 G    C$BSUB= 000002       DEV6    005040      F$SEG = 000003      I$AUTO= 000041
BIT13 = 020000 G    C$CEFG= 000045       D V7    005250      F$SOFT= 000005      I$CLN = 000041
BIT14 = 040000 G    C$CLCK= 000062       DFPTBL  002172 G    F$SRV = 000010      I$DU  = 000041
BIT15 = 100000 G    C$CLEA= 000012       DFTL1   012206      F$SUB = 000002      I$HRD = 000041
BIT2  = 000004 G    C$CLOS= 000035       DIAGMC= 000000      F$SW  = 000014      I$INIT= 000041
BIT3  = 000010 G    C$CLP1= 000006       DLV   = 000074 G    F$TEST= 000001      I$MOD = 000041
BIT4  = 000020 G    C$CVEC= 000036       DO RK   013222 G    GBIMP   010106      I$MSG - 000041
BIT5  = 000040 G    C$DCLN= 000044       DONE    003314 G    GBIMP2  010110      I$PROT= 000040
BIT6  = 000100 G    C$DODU= 000051       DR    = 000060 G    GETANS  006736 G    I$PTAB= 000041
BIT7  = 000200 G    C$DRPT= 000024       DRVCHK  002214      GETHRD  015524      I$PWR = 000041
BIT8  = 000400 G    C$DU  = 000053       EF.CON= 000036 G    GETPTR  007002      I$RPT = 000041
BIT9  = 001000 G    C$EDIT= 000003       EF.NEW= 000035 G    GETR5   016424      I$SEG   000041
BLKEND= 000202 G    C$ERDF= 000055       EF.PWR= 000034 G    GTAGIN  007224      I$SETU= 000041
BLKER   003324 G    C$ERHR= 000056       EF.RES= 000037 G    GTBYTE  007662 G    I$SFT = 000041
BLKSIZ= 000210 G    C$ERRO= 000060       EF.STA= 000040 G    GTDOWN  007534      I$SRV = 000041
BLKTBL  003340 G    C$ERSF= 000054       ERRDES  012402 G    GTOK    007452      I$SUB = 000041
BOE   = 000400 G    C$ERSO= 000057       ESABO = 177720 G    GTPKS1  0070?4 G    I$TSI = 000041
BRKPTR  013760      C$ESCA= 000010       ESCKS = 177757 G    GTPKS8  0072 4 G    J$JMP = 000167
BRKTO   013756      C$ESEG= 000005       ESCKSM= 177757      GTPTR   00766?      LENGTH  002204
BRKWD   013752      C$ESUB= 000003       ESCMD = 177720 G    GTUM    007414      LGOFST= 000120 G
BUFTBL  024346      C$ETST= 000001       ESNCRT= 177767 G    G$CNTO= 000?00      LNCNT   014342
BUF0    025426      C$EXIT= 000032       ESNOMO= 177737 G    G$DELM= 000372      LOE   = 040000 G
BUF1    026464      C$GETB= 000026       ESNONX= 177770 G    G$DISP= 000003      LOG     012046 G
BUF2    027522      C$GETW= 000027       ESOK  = 000000 G    G$EXCP= 000400      LOGO    012370
BUF3    030560      C$GMAN= 000043       ESPART= 177776 G    G$HILI= 000002      LOGOK   012134
BUF4    031616      C$GPHR= 000042       ESRD  = 177757      G$LOLI= 000001      LOGOK2  012220
BUF5    032654      C$GPLO= 000030       ESREC = 177711 G    G$NO  = 000000      LOGO    012122
BUF6    033712      C$GPRI= 000040       ESSK  = 177740 G    G$OFFS= 000400      LOG1    012236
BUF7    034752      C$INIT= 000011       ESSLF = 177777 G    G$UFSI= 000376      LOG2    012266
CARLF   014356      C$INLP= 000020       ESTRY = 000001 G    G$PRMA= 000001      LOG3    012326
CHECK   015502      C$MANI= 000050       ESWLOC= 177765 G    G$PRMD= 000002      LOG3B   012350
CHKANR  010176      C$MEM = 000031       ESWR  = 177757      G$PRML= 000000      LOI   = 000010 G
```

```
L$TDEV  003356 G    L10004  013660      NCAR1 = 000054 G    RECERR= 000042 G    STRT    016150 G
L$ACP   002110 G    L10005  013714      NODRVS  016116     RECID   012654 G    SUCCS = 000076 G
L$APT   002036 G    L10006  015004      NOMCR   006340     RECID2  013030      SUCOTL= 000046 G
L$AU    016504 G    L10007  016064      NOMOT = 000030 G    RECOV   011262      SVCGBL= 000000
L$AUT   002070 G    L10010  016262      NOUNIT= 000036 G    RETERR  011464      SVCINS= 177777
L$AUTO  016156 G    L10011  016356      NOXOFF  006430     RETRY = 000002 G    SVCSUB= 177777
L$CCP   002106 G    L10012  016422      NTSFT   012250     RLUN    015006      SVCTAG= 177777
L$CLEA  016340 G    L10013  016504      NXTRET  006334     RPTR    015010      SVCTST= 177777
L$CO    002032 G    L10014  016703      NXTST   006052 G   RSCMND= 000002 G    SWAPDR  005500 G
L$DEPO  002011 G    L10015  017144      NXTST2  006156     RSCONT= 000020 G    SWPTR   005600
L$DESC  002122 G    L10016  020534      OVERFL= 000001      RSDASZ= 000204 G    SYSTAT  003300 G
L$DESP  002076 G    L10017  021510      OTL   = 000052 G    RSDATA= 000001 G    S$LSYM= 010000
L$DEVP  002060 G    L10020  022274      OVRFLO  012746     RSDNSZ= 000222 G    TAPLEN  003302 G
L$DISP  002152 G    L10021  023250      OVRN  = 000012 G    RSEND = 000002 G    THRSHI  011370
L$DLY   002116 G    L10022  024034      O$APTS= 000000      RSINIT= 000004 G    THRSLO  011342
L$DTP   002040 G    L10023  036052      O$AU  = 000001      RSMSIZ= 000012 G    TMP   = 000066 G
L$DTYP  002034 G    L10024  036242      O$BGNR= 000001      RSNDSZ= 000016 G    TOMANY  016066
L$DU    016360 G    L10025  036564      O$BGNS= 000001      RSNTAB  002220      TORCVB= 000050 G
L$DUT   002072 G    L10027  036574      O$DU  = 000001      RSSEND= 000100 G    TOSNDB= 000056 G
L$DVTY  005460 G    MABEE   012300      O$ERRT= 000000      RSSNIT= 000001 G    TRBUF   024370
L$EF    002052 G    MSAUTO  016320      O$GNSW= 000001      RSSNOP= 000000 G    TRK   = 000062 G
L$ENV1  002044 G    MSBDA   002332 G    O$POIN= 000001      RSSNSZ= 000016 G    TRPHND  016266
L$ETP   002102 G    MSCMD   002676 G    O$SETU= 000001      RSSRD = 000002 G    TSPPTR  016254
L$EXP1  002046 G    MSCOM   002376 G    PARTL = 000034 G    RSSSEK= 000005 G    TSTPC = 000020 G
L$EXP4  002064 G    MSG1    036052      PATTEN= 000072 G    RSSSLF= 000007 G    TSTTOP  003320
L$EXP5  002066 G    MSG1B   036063      PDTFLG  016152 G   RSSWR = 000003 G    TST1    016552
L$HARD  036010 G    MSG1C   036100      PERDEV  006170     RSVP    006364 G    TST2    016746
L$HIME  002120 G    MSG2    036131      PKPTR = 000104 G    RSXOFF= 000023 G    TST3    017212
L$HPCP  002016 G    MSG3    036146      PNT   = 001000 G    RSXON = 000021 G    TST3PT  020522
L$HPTP  002022 G    MSG4    036242      PRBUF   002210      RTRYN   011422      TST4    020602
L$HW    002172 G    MSG4B   036307      PRDAT   014344      RUN     006022 G    TST4EX  021502
L$ICP   002104 G    MSG5    036351      PRFORM  014346      SECREC  003326 G    TST5    021556
L$INIT  015376 G    MSG6    036403      PRI   = 002000 G    SERVST  007656      TST5EX  022266
L$LADP  002026 G    MSG7    036430      PRI00 = 000000 G    SETDR   005602 G    TST6    022342
L$LAST  036560 G    MSG8    036456      PRI01 = 000040 G    SETLEN  016026      TST6EX  023242
L$LOAD  002100 G    MSHCHK  002550 G    PRI02 = 000100 G    SETPTR  005656      TST7    023316
L$LUN   002074 G    MSHDRD  003146 G    PRI03 = 000140 G    SETSRV  007572      TST7EX  024026
L$MREV  002050 G    MSHDWR  003210 G    PRI04 = 000200 G    SETUP   005754 G    TUVECT= 000204 G
L$NAME  002000 G    MSNIT   002612 G    PRI05 = 000240 G    SFPTBL  002204 G    T$ARGC= 000002
L$PRIO  002042 G    MSNLOG  002314 G    PRI06 = 000300 G    SFT     012316      T$CODE= 005052
L$PROT  002142 G    MSNOMO  002440 G    PRI07 = 000340 G    SFTOUT  036242      T$ERRN= 000146
L$PRT   002112 G    MSNOTP  002456 G    PRNPAK  014176 G   SFTRD = 000002 G    T$EXCP= 000000
L$REPP  002062 G    MSNRSP  002756 G    PRNSIZ  003330 G   SFTWR = 000004 G    T$FLAG= 000040
L$REV   002010 G    MSOVRN  003252 G    PTR     016454      SKERR = 000024 G    T$FREE= 036574
L$RPT   014362 G    MSPART  002626 G    RCBCNT  003310      SLFER = 000044 G    T$GMAN= 000000
L$SOFT  036166 G    MSQRSP  002772 G    RCBFSZ= 001036 G    SND     006434      T$HILI= 000376
L$SPC   002056 G    MSREC   002712 G    RCDB  = 000024 G    SNDBYT  006666 G    T$LAST= 000001
L$SPCP  002020 G    MSRNIT  002530 G    RCFLG   003306 G    SNDCNT= 000070 G    T$LOLI= 000001
L$SPTP  002024 G    MSSELF  002356 G    RCVNIT= 000006 G    SNDHND  013646      T$LSYM= 010000
L$STA   002030 G    MSSFRD  003046 G    RCSR  = 000022 G    SNDINT  013646 G    T$LTNO= 000007
L$SW    002204 G    MSSFWR  003106 G    RCVBUF= 000102 G    SOFTR = 000122 G    T$NEST= 177777
L$TEST  002114 G    MSSKER  002300 G    RCVHND  013662      SOFTW = 000124 G    T$NS0 = 000000
L$TIML  002014 G    MSTOSN  003024 G    RCVINT  013662 G    SRVTBL  007636      T$NS1 = 000005
L$UNIT  002012 G    MSUNIT  002650 G    RDNO  = 000114 G    STAEOP  002206      T$PCNT= 000000
L10001  002202      MSWPRO  002506 G    RDN1  = 000116 G    STATHD  015012      T$PTAB= 010026
L10002  002220      MSWRSP  002732 G    REC   = 000064 G    STATUS= 000000 G    T$PTHV= 000001
L10003  012572      MXRTRY  003322 G    RECDAT  017130      STHD2   015266      T$PTNO= 000001
```

```
T$SAVL= 177777          T$$CLE= 010011          T$$SRV= 010005          T7      023252 G        XFNSND  006416
T$SEGL= 177777          T$$DAT= 010027          T$$SW = 010002          UAM   = 000200 G        XMDB  = 000030 G
T$SIZE= 000006          T$$DU = 010012          T$$TES= 010022          UNIT    012574 G        XMSR  = 000026 G
T$SUBN= 000000          T$$HAR= 010023          T1      016506 G        UNREC   011442          XSCNT = 000036 G
T$TAGL= 177777          T$$HW = 010001          T1TRY = 000146 G        UNSUC   011040          XSFLG = 000034 G
T$TAGN= 010030          T$$INI= 010007          T2      016702 G        UNXPCT  007356          XSPKNM= 000032 G
T$TEMP= 000000          T$$MSG= 010003          T3      017146 G        WAIT    013716          XSPTR = 000106 G
T$TEST= 000007          T$$PC = 000001          T4      020536 G        WHCHDR  013052 G        X$ALWA= 000000
T$TSTM= 177777          T$$PRO= 010000          T4TRY = 000132 G        WRLOCK= 000026 G        X$FALS= 000040
T$TSTS= 000001          T$$PTA= 010026          T5      021512 G        WRIN0 = 000110 G        X$OFFS= 000400
T$$AU = 010013          T$$RPT= 010006          T6      022276 G        WRIN1 = 000112 G        X$TRLE= 00002C
T$$AUT= 010010          T$$SOF= 010024

. ABS.  036574     000
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  33125 WORDS  ( 130 PAGES)
DYNAMIC MEMORY:  20308 WORDS  ( 78 PAGES)
ELAPSED TIME:  00:06:55
CZTUUB.BIN/EN:AMA:ABS,CZTUUB/CR/-SP LB1:[1,1]SVC/MLB,SY:[203,230]CZTUUB.MAC
```

SYMBOL	VALUE		REFERENCES								
ABNDX	= 000004	G	#26-819	*84-2428	84-2441	84-2443					
ABO	012360		84-2438	84-2452	84-2465	#84-2481					
ABOMSG	016456		109-3180	#109-3210							
ABONM	006336		*60-1626	*60-1643	60-1648	#60-1656					
ADR	= 000020	G	#14-567								
ALLGON	003332	G	#24-787	*60-1652	*103-3045	107-3137					
ASSEMB	= 000010		5-375	5-375							
BDATA	= 000134	G	#28-872	101-2922	101-2932						
BDBYTS	014140		*98-2770	*98-2781	98-2784	98-2789	#98-2799				
BDCHK	= 000022	G	#16-595	78-2146	78-2198						
BDCOM	= 000014	G	#16-592	80-2242	84-2453						
BIT0	= 00 001	G	#14-567	48-1467	72-1987	84-2444	90-2555	90-2556	90-2558	90-2569	94-2644
BIT00	= 000001	G	#14-567	14-567							
BIT01	000002	G	#14-567	14-567							
BIT02	= 000004	G	#14-567	14-567							
BIT03	= 000010	G	#14-567	14-567							
BIT04	= 000020	G	#14-567	14-567							
BIT05	= 000040	G	#14-567	14-567							
BIT06	= 000100	G	#14-567	14-567							
BIT07	= 000200	G	#14-567	14-567							
BIT08	= 000400	G	#14-567	14-567							
BIT09	= 001000	G	#14-567	14-567							
BIT1	= 000002	G	#14-567	60-1617	66-1793	72-1988	76-2087	80-2230	80-2299		
BIT10	= 002000	G	#14-567	60-1613	62-1723	80-2248	80-2268	80-2276	80-2297	118-3363	118-3363
			118-3364	120-3406	120-3406	122-3437	124-3468	124-3468	126-3499		
BIT11	= 004000	G	#14-567	101-2908	103-2983	103-3004	105-3104				
BIT12	= 010000	G	#14-567	118-3363	118-3363	118-3363	120-3406	120-3406	120-3406	124-3468	124-3468
			124-3468								
BIT13	= 020000	G	#14-567	103-2982	103-3029	109-3177					
BIT14	= 040000	G	#14-567	60-1632	60-1635	78-2159	78-2171	94-2694	103-3010		
BIT15	= 100000	G	#14-567	48-1465	68-1835	68-1857	70-1893	70-1910	70-1930	70-1941	76-2095
			80-2234	103-2982	103-3029	105-3104	109-3177				
BIT2	= 000004	G	#14-567	72-1989							
BIT3	= 000010	G	#14-567	72-1990							
BIT4	= 000020	G	#14-567	62-1717	62-1721	72-1991	80-2500	94-2646			
BIT5	= 000040	G	#14-567	62-1739	62-1741	72-1992					
BIT6	= 000100	G	#14-567	72-1993	80-2239	80-2300	94-2671	94-2679	94-2686	94-2702	94-2705
			96-2715	96-2723	98-2790						
BIT7	= 000200	G	#14-567	72-1994	80-2268	80-2276	80-2282	80-2284	80-2286	80-2297	
BIT8	= 000400	G	#14-567	50-1500	62-1727	62-1736	80-2261	80-2290	105-3011		
BIT9	= 001000	G	#14-567	48-1469	62-1711	80-2245	80-2272	82-2355	105-3013		
BLKEND	= 000202	G	#28-892	103-3021							
BLKER	003324	G	#24-784	*62-1708	*103-3053						
BLKSIZ	= 000210	G	#28-896	30-918	30-919	30-920	30-921	30-922	30-923	30-924	30-925
BLKTBL	003340	G	#30-906	48-1463	50-1497	52-1522	56-1565	60-1611	60-1627	70-1897	76-2095
			101-2902	103-2978	103-2998	105-3102	109-3201				
BOE	= 000400	G	#14-567								
BRKPTR	013760		*94-2670	*94-2678	*94-2685	96-2736	#96-2750				
BRKTO	013756		*94-2643	*94-2652	*94-2658	*94-2668	*94-2676	*96-2740	#96-2749		
BRKWD	013752		94-2657	#96-2746							
BUFTBL	024346		105-2985	#130-3526							
BUFO	025426		130-3526	# 30-3546							

SYMBOL	VALUE		REFERENCES								
BUF1	026464		130-3527	#130-3547							
BUF2	027522		130-3528	#130-3548							
BUF3	030560		130-3529	#130-3549							
BUF4	031616		130-3530	#130-3550							
BUF5	032654		130-3531	#130-3551							
BUF6	033712		130-3532	#130-3552							
BUF7	034750		130-3533	#130-3553							
CARLF	014356		100-2828	100-2830	#100-2839						
CHECK	015502		103-2988	#103-2993							
CHKANR	010176		76-2091	76-2099	#76-2103						
CHKANS	010112	G	58-1590	#76-2085							
CHKEND	010536	G	78-2151	78-2194	#80-2228						
CHKERR	010624		80-2238	#80-2245							
CHKPKS	010202	G	76-2089	76-2097	#78-2126						
CHKPTR	010200		•76-2093	76-2094	76-2098	•76-2100	#76-2105				
CHKREE	010676		80-2231	#80-2252							
CHKRET	011242		80-2236	80-2241	80-2244	80-2247	80-2251	80-2269	80-2277	80-2285	80-2287
			#80-2299								
CHKSUC	011536	G	80-2232	80-2252	#82-2334						
CHKSUM	013066	G	#90-2553	92-2596	114-3314	116-3332	118-3363	118-3363	118-3363	118-3364	118-3364
			120-3406	120-3406	120-3406	122-3437	122-3437	124-3468	124-3468	124-3468	126-3499
			126-3499								
CHKB	010132		76-2088	#76-2093							
C CKSM	013162	G	78-2144	78-2179	78-2192	#92-2593					
CLRALL	005660	G	#52-1522	66-1796							
CLRBUF	005720	G	52-1524	#54-1540	66-1799						
CLRPTR	005752		•52-1522	52-1523	52-1525	•52-1527	#54-1550				
CMDSNT	= 000100	G	#26-844	•62-1720	82-2341	82-2345	82-2358	86-2506	•94-2645		
CMNDER	= 000040	G	#16-602	82-2378							
CMPDAT	002212		#13-494	98-2772							
CNINIT	= 000032	G	#16-599	94-2698							
COMPAR	013762	G	78-2156	78-2181	#98-2767						
CSNRDY	003334	G	#24-794	64-1764							
CSRCVB	003336	G	#24-795	74-2024							
C$$U	= 000052		#5-375	111-3245							
C$AUTO	= 000061		#5-375	105-3113							
C$BRK	= 000022		#5-375	58-1588	60-1651	64-1768	74-2046	94-2648	96-2738	96-2739	101-2901
			101-2904	101-2906							
C$BSEG	= 000004		#5-375								
C$BSUB	= 000002		#5-375								
C$CEFG	= 000045		#5-375								
C$CLCK	= 000062		#5-375								
C$CLEA	= 000012		#5-375	107-3162							
C$CLOS	= 000035		#5-375								
C$CLPT	= 000006		#5-375								
C$CVEC	= 000016		#5-375	94-2704	94-2706	105-3112					
C$DCLN	= 000044		#5-375	60-1653	103-2996	103-3016					
C$DODU	= 000051		#5-375	84-2483	105-3124						
C$DRPT	= 000024		#5-375	107-3141							
C$DU	= 000053		#5-375	109-3200							
C$EDIT	= 000003		#5-375	5-418							
C$ERDF	= 000055		#5-375	84-2437	84-2451						

SYMBOL	VALUE		REFERENCES							
C$ERHR	= 000056		#5-375	84-2464	84-2479					
C$ERRO	= 000060		#5-375							
C$ERSF	= 000054		#5-375	60-1650	103-2995	103-3015				
C$ERSO	= 000057		#5-375	84-2457	84-2475	98-2788				
C$ESCA	= 000010		#5-375							
C$ESEG	= 000005		#5-375							
C$ESUB	= 000003		#5-375							
C$ETST	= 000001		#5-375	114-3319	116-3346	118-3388	120-3420	122-3451	124-3482	126-3513
C$EXIT	= 000032		#5-375	114-3312	116-3326	118-3356	120-3396	122-3427	124-3458	126-3489
C$GETB	= 000026		#5-375							
C$GETW	= 000027		#5-375							
C$GMAN	= 000043		#5-375							
C$GPHR	= 000042		#5-375	103-3002						
C$GPLO	= 000030		#5-375							
C$GPRI	= 000040		#5-375							
C$INIT	= 000011		#5-375	103-3082						
C$INLP	= 000020		#5-375							
C$MANI	= 000050		#5-375							
C$MEM	= 000031		#5-375							
C$MSG	= 000023		#5-375	86-2513						
C$OPEN	= 000034		#5-375							
C$PNTB	= 000014		#5-375	80-2274	86-2503	86-2506	86-2509	98-2789		
C$PNTF	= 000017		#5-375	100-2823	100-2828	100-2830	105-3121	109-3180		
C$PNTS	= 000016		#5-375	101-2903	101-2905	101-2921	101-2922	101-2932		
C$PNTX	= 000015		#5-375	80-2250	80-2255	80-2258	80-2260	80-2281	90-2289	
C$Q10	= 000377		#5-375							
C$RDPU	= 000007		#5-375							
C$REFG	= 000047		#5-375	103-2975						
C$RESE	= 000033		#5-375	#5-375						
C$REVI	= 000003		#5-375	5-418						
C$RFLA	= 000021		#5-375	103-3052						
C$RPT	= 000025		#5-375	101-2945						
C$SEFG	= 000046		#5-375							
C$SPRI	= 000041		#5-375	94-2663						
C$SVEC	= 000037		#5-375	94-2664	94-2666	105-3101				
C$TPRI	= 000013		#5-375							
DESC	014142		98-2789	#98-2800						
DEVPTR	003304	G	#24-776	•60-1611	60-1612	60-1621	•60-1623	•60-1627	60-1628	60-1644	•60-1646
			•103-2978	103-2980	103-2987	•103-2989	•103-2998	103-3000	•103-3046		
DEV0	003360		30-906	#30-918						
DEV1	003570		30-907	#30-919						
DEV2	004000		30-908	#30-920						
DEV3	004210		30-909	#30-921						
DEV4	004420		30-910	#30-922						
DEV5	004630		30-911	#30-923						
DEV6	005040		30-912	#30-924						
DEV7	005250		30-913	#30-925						
DFPTBL	02172	G	#11-466							
DFTL1	012206		•84-2449	•84-2450	#84-2451	84-2451				
DIAGMC	= 000000		5-375	5-375						
DLV	= 000074	G	#26-842	68-1855	70-1932	70-1943	•74-2027	74-2028	74-2029	•74-2031	•74-2052
			74-2053	74-2054	•74-2056	78-2161	86-2507	86-2509	•86-2510	•96-2724	96-2725

SYMBOL	VALUE		REFERENCES								
			96-2726	•96-2728	•103-3044						
DOBRK	013222	G	60-1634	#94-2642							
DONE	003314	G	#24-780	•56-1564	58-1584	•114-3315	•116-3327	•118-3379	•120-3417	•122-3448	•124-3479
			•126-3510								
DR	= 000060	G	#26-835	48-1467	48-1469	•48-1471	•50-1499	50-1500	•50-1502	86-2503	88-2534
			•103-3007	103-3011	103-3013	116-3332	118-3363	118-3363	118-3364	118-3364	120-3405
			120-3406	120-3406	122-3436	122-3437	122-3437	124-3467	124-3468	124-3468	126-3498
			126-3499	126-3499							
DRVCHK	002214		#13-495	120-3403	122-3434	124-3465	126-3496				
EF.CON	= 000036	G	#14-567								
EF.NEW	= 000035	G	#14-567								
EF.PWR	= 000034	G	#14-567								
EF.RES	= 000037	G	#14-567								
EF.STA	= 000040	G	#14-567	103-2975							
ERRDES	012402	G	84-2437	84-2451	84-2457	84-2464	84-2475	84-2479	#86-2496	98-2788	
ESABO	= 177720	G	#18-661								
ESCKS	= 177757	G	#18-672	18-674	18-675	18-676	82-2353				
ESCKSM	= 177757		#18-674								
ESCMD	= 177720	G	#18-670	82-2376							
ESNCRT	= 177767	G	#18-662	82-2371							
ESNOMO	= 177737	G	#18-669	82-2348							
ESNONX	= 177770	G	#18-663	82-2381							
ESOK	= 000000	G	#18-664	82-2336							
ESPART	= 177776	G	#18-665	82-2391							
ESRD	= 177757		#18-672								
ESREC	= 177711	G	#18-671	82-2396							
ESSK	= 177740	G	#18-666	82-2366							
ESSLF	= 177777	G	#18-673								
ESTRY	= 000001	G	#18-667	82-2339							
ESWLOC	= 177765	G	#18-668	82-2386							
ESWR	= 177757		#18-675								
EVL	= 000004	G	#14-567	84-2445							
EVLTMR	002216		#13-496	84-2447							
EXOFF	007213		#68-1870	70-1950	74-2039						
EXON	007212		68-1826	#68-1869	70-1905	74-2047	74-2059				
FSEND	= 002100		#5-375								
FSLOAD	= 000035		#5-375	5-418							
FLGLOC	016154	G	84-2445	•103-3052	#103-3091						
FM	015142		101-2922	101-2932	#101-2955						
FMO	015124		101-2921	#101-2952							
FTLNM	003312		#24-779	84-2468							
FSAU	= 000015		#5-375	111-3222	111-3245						
FSAUTO	= 000020		#5-375	105-3099	105-3113						
FSBGN	= 000040		#5-375	5-401	7-428	13-507	14-560	86-2496	96-2713	96-2721	100-2842
			101-2886	101-2893	101-2961	103-2971	105-3099	107-3136	109-3172	111-3222	114-3307
			114-3310	114-3312	114-3319	116-3324	116-3326	116-3346	118-3354	118-3354	118-3388
			120-3394	120-3396	120-3420	122-3425	122-3427	122-3451	124-3456	124-3458	124-3482
			126-3487	126-3489	126-3513	130-3557	132-3622	132-3635	134-3677	136-3714	136-3716
			136-3717	136-3717	136-3722	136-3723					
FSCLEA	- 000007		#5-375	107-3136	107-3162						
FSDU	= 000016		#5-375	109-3172	109-3200						
FSEND	- 000041		#5-375	5-375	5-375	5-375	5-375	5-375	5-375	5-375	5-375

SYMBOL	VALUE		REFERENCES								
			5-375	5-375	5-375	5-375	5-375	5-375	5-375	5-375	5-401
			13-507	14-560	86-2513	96-2717	96-2730	100-2842	101-2886	101-2945	101-2961
			103-3082	105-3113	107-3162	109-3200	111-3245	114-3307	114-3310	114-3310	114-3310
			114-3312	114-3319	114-3319	116-3324	116-3324	116-3324	116-3326	116-3346	116-3346
			118-3354	118-3354	118-3354	118-3356	118-3388	118-3388	120-3394	120-3394	120-3394
			120-3396	120-3420	120-3420	122-3425	122-3425	122-3425	122-3427	122-3451	122-3451
			124-3456	124-3456	124-3456	124-3458	124-3482	124-3482	126-3487	126-3487	126-3487
			126-3489	126-3513	126-3513	130-3557	132-3622	132-3648	134-3693	136-3714	136-3716
			136-3717	136-3722	136-3723						
F$HARD	= 000004		#5-375	132-3633	132-3648						
F$HW	= 000013		#5-375	11-466	11-479						
F$INIT	= 000006		#5-375	103-2971	103-3082						
F$JMP	= 000050		#5-375	114-3312	116-3326	118-3356	120-3396	122-3427	124-3458	126-3489	
F$MOD	= 000000		#5-375	5-401	13-507	14-560	100-2842	101-2886	101-2961	114-3307	130-3557
			132-3622	136-3714							
F$MSG	= 000011		#5-375	86-2496	86-2513						
F$PROT	= 000021		#5-375	7-428	7-432						
F$PWR	= 000017		#5-375								
F$RPT	= 000012		#5-375	101-2893	101-2945						
F$SVC	= 000003		#5-375								
F$SUFT	= 000005		#5-375	134-3677	134-3693						
F$SRV	= 000010		#5-375	96-2713	96-2717	96-2721	96-2730				
F$SUB	= 000002		#5-375								
F$SW	= 000014		#5-375	13-489	13-505						
F$TEST	= 000001		#5-375	114-3310	114-3319	116-3324	116-3346	118-3354	118-3388	120-3394	120-3420
			122-3425	122-3451	124-3456	124-3482	126-3487	126-3513			
GBTMP	010106		*74-2023	*74-2033	*74-2043	74-2044	*74-2063	#74-2069			
GBTMP2	010110		*74-2038	74-2049	74-2051	*74-2058	74-2061	#74-2070			
GETANS	006736	G	58-1586	#66-1792							
GETMRD	015524		103-2994	#103-2998							
GETPTR	007002		#66-1804								
GETRS	016424		109-3176	#109-3201							
GTAGIN	007224		#70-1892	70-1956							
GTBYTE	007662	G	68-1834	68-1854	70-1909	70-1929	70-1940	#74-2023			
GTDOWN	007534		70-1896	70-1900	70-1913	70-1924	70-1928	70-1931	70-1935	70-1942	70-1946
			70-1948	#70-1952							
GTOK	007452		70-1917	#70-1937							
GTPKS1	007004	G	66-1801	#68-1821							
GTPKS8	007214	G	66-1797	#70-1890	70-1959						
GTPTR	007660		*70-1891	70-1892	70-1953	*70-1955	#72-1997				
GTUM	007414		70-1922	#70-1926							
GSCNTO	= 000200		#5-375								
GSDELM	= 000372		#5-375								
GSDISP	= 000003		#5-375								
GSEXCP	= 000400		#5-375								
GSHILI	= 000002		#5-375								
GSLOLI	= 000001		#5-375								
GSNO	= 000000		#5-375								
GSOFFS	= 000400		#5-375	132-3636	132-3637	132-3638	132-3639	132-3640	134-3679	134-3680	134-3681
			134-3682	134-3683	134-3685						
GSOFS1	= 000376		#5-375	132-3636	132-3637	132-3638	132-3639	132-3640	134-3679	134-3680	134-3681
			134-3682	134-3683	134-3685						

SYMBOL	VALUE		REFERENCES								
G$PRMA	= 000001		#5-375	132-3636	132-3637						
G$PRMD	= 000002		#5-375	134-3679	134-3685						
G$PRML	= 000000		#5-375	132-3638	132-3639	132-3640	134-3680	134-3681	134-3682	134-3683	
G$RADA	= 000140		#5-375								
G$RADB	= 000000		#5-375								
G$RADD	= 000040		#5-375	134-3679	134-3685						
G$RADL	= 000120		#5-375	132-3638	132-3639	132-3640	134-3680	134-3681	134-3682	134-3683	
G$RADO	= 000020		#5-375	132-3636	132-3637						
G$XFER	= 000004		#5-375								
G$YES	= 000010		#5-375	132-3636	132-3637	132-3638	132-3639	132-3640	134-3679	134-3680	134-3681
			134-3682	134-3683	134-3685						
HARDR	= 000136	G	#28-873	101-2917	101-2927						
HARDW	= 000140	G	#28-874	101-2919	101-2929						
HELP	= 000000		#5-360	5-370	5-392	5-410	9-435	9-450	11-473	13-498	#14-512
			14-550	14-569	30-926	30-932	32-948	32-953	32-961	32-968	32-973
			32-979	48-1395	48-1407	48-1412	48-1418	48-1423	48-1429	48-1437	48-1444
			48-1450	48-1456	#101-2848	103-3060	103-3070	107-3143	107-3150	109-3182	109-3188
			111-3225	111-3231	#112-3251	112-3292	112-3298	130-3558	130-3563	130-3573	#132-3584
			132-3642	132-3658	134-3686	136-3707					
HOE	= 100000	G	#14-567								
HRD	012340		84-2469	#84-2477							
HPDRD	= 000016	G	#16-593	80-2292							
HRDWR	= 000020	G	#16-594	80-2294							
HRD1	011164		80-2279	#80-2288							
IBE	= 010000	G	#14-567								
IDPTR	003316	G	#24-781	*56-1567	56-1566	56-1568	*56-1570				
IDU	= 000040	G	#14-567								
IER	= 020000	G	#14-567								
INIT	015376		#103-2973								
INITWD	013754		*94-2642	94-2669	94-2677	94-2684	94-2691	94-2697	#96-2747		
INIT2	015420		103-2976	#103-2978							
ISR	= 000100	G	#14-567								
IXE	= 004000	G	#14-567								
ISAU	= 000041		#5-375	#111-3222	#111-3245						
ISAUTO	= 000041		#5-375	#105-3099	#105-3113						
ISCLN	= 000041		#5-375	#107-3136	#107-3162						
ISDU	= 000041		#5-375	#109-3172	#109-3200						
ISHRD	= 000041		#132-3643	#132-3648							
ISINIT	= 000041		#5-375	#103-2971	#103-3082						
ISMOD	= 000041		#5-375	5-401	#5-401	13-507	#13-507	14-560	#14-560	100-2842	#100-2842
			101-2886	#101-2886	101-2961	#101-2961	114-3307	#114-3307	130-3557	#130-3557	132-3622
			#132-3622	136-3714	#136-3714						
ISMSG	= 000041		#5-375	#86-2496	#86-2513						
ISPROT	= 000040		#5-375	#7-428							
ISPTAB	= 000041		#5-375	136-3717	#136-3717	136-3722	#136-3722				
ISPWR	= 000041		#5-375								
ISRPT	= 000041		#5-375	#101-2893	#101-2945						
ISSEG	= 000041		#5-375	114-3310	116-3324	118-3354	120-3394	122-3425	124-3456	126-3487	
ISSETU	= 000041		#5-375	136-3716	#136-3716	136-3717	136-3723	#136-3723			
ISSFT	= 000041		#134-3677	#134-3693							
ISSRV	= 000041		#5-375	#96-2715	#96-2717	#96-2721	#96-2730				
ISSUB	000041		#5-375	114-3310	116-3324	118-3354	120-3394	122-3425	124-3456	126-3487	

SYMBOL	VALUE		REFERENCES								
I$TST	= 000041		#5-375	114-3310	#114-3310	114-3312	114-3319	#114-3319	#114-3319	116-3324	#116-3324
			116-3326	116-3346	#116-3346	#116-3346	118-3354	#118-3354	118-3356	118-3388	#118-3388
			#118-3388	120-3394	#120-3394	120-3396	120-3420	#120-3420	#120-3420	122-3425	#122-3425
			122-3427	122-3451	#122-3451	#122-3451	124-3456	#124-3456	124-3458	124-3482	#124-3482
			#124-3482	126-3487	#126-3487	126-3489	126-3513	#126-3513	#126-3513		
J$JMP	= 000167		#5-375								
LENGTH	002204		#13-491	103-3054							
LGOFST	= 000120	G	#28-863	84-2429							
LNCNT	014342		*100-2821	*100-2826	#100-2835						
LOE	= 040000	G	#14-567								
LOG	012046	G	64-1774	74-2067	78-2147	78-2164	78-2173	78-2199	80-2243	80-2266	80-2295
			82-2402	#84-2421	94-2655	94-2699	96-2743				
LOGO	012370		84-2458	84-2476	#84-2484						
LOGOK	012134		84-2436	#84-2439							
LOGOK2	012220		84-2446	84-2448	#84-2453						
LOGO	012122		#84-2437								
LOG1	012236		*84-2455	*84-2456	#84-2457	84-2457					
LOG2	012266		*84-2462	*84-2463	#84-2464	84-2464					
LOG3	012326		*84-2473	*84-2474	#84-2475	84-2475					
LOG3B	012350		*84-2477	*84-2478	#84-2479	84-2479					
LOT	= 000010	G	#14-567								
LSTDEV	003356	G	#30-913	48-1473	50-1503	52-1525	56-1568	60-1621	60-1644	70-1953	76-2098
			101-2933	103-2987	105-3108						
L$ACP	002110	G	#5-418								
L$APT	002036	G	#5-418								
L$AU	016504	G	5-418	#111-3222							
L$AUT	002070	G	#5-418								
L$AUTO	016156	G	5-418	#105-3099							
L$CCP	002106	G	#5-418								
L$CLEA	016340	G	5-418	#107-3136							
L$CO	002032	G	#5-418								
L$DEPO	002011	G	#5-418								
L$DESC	002122	G	5-418	#5-420							
L$DESP	002076	G	#5-418								
L$DEVP	002060	G	#5-418								
L$DISP	002152	G	5-418	#9-448							
L$DLY	002116	G	#5-418								
L$DTP	002040	G	#5-418								
L$DTYP	002034	G	#5-418								
L$DU	016360	G	5-418	#109-3172							
L$DUT	002072	G	#5-418								
L$DVTY	005460	G	5-418	#32-946							
L$EF	002052	G	#5-418								
L$ENVI	002044	G	#5-418								
L$ETP	002102	G	#5-418								
L$EXP1	002046	G	#5-418								
L$EXP4	002064	G	#5-418								
L$EXP5	002066	G	#5-418								
L$HARD	036010	G	5-418	132-3633	#132-3633						
L$HIME	002120	G	#5-418								
L$HPCP	002016	G	#5-418								
L$HPTP	002022	G	#5-418								

SYMBOL	VALUE		REFERENCES				
L$HW	002172	G	5-418	11-466	#11-466		
L$ICP	002104	G	#5-418				
L$INIT	015376	G	5-418	#103-2971			
L$LADP	002026	G	#5-418				
L$LAST	036560	G	5-418	#136-3713	136-3723		
L$LOAD	002100	G	#5-418				
L$LUN	002074	G	#5-418	•84-2426	•84-2427	•98-2786	•98-2787 •103-3001
L$MREV	002050	G	#5-418				
L$NAME	002000	G	#5-418				
L$PRIO	002042	G	#5-418				
L$PROT	002142	G	5-418	#7-428			
L$PRT	002112	G	#5-418				
L$REPP	002062	G	#5-418				
L$REV	002010	G	#5-418				
L$RPT	014362	G	5-418	#101-2893			
L$SOFT	036166	G	5-418	134-3677	#134-3677		
L$SPC	002056	G	#5-418				
L$SPCP	002020	G	#5-418				
L$SPTP	002024	G	#5-418				
L$STA	002030	G	#5-418				
L$SW	002204	G	5-418	13-489	#13-489		
L$TEST	002114	G	#5-418				
L$TIML	002014	G	#5-418				
L$UNIT	002012	G	#5-418	103-2993	103-3048		
L10001	002202		11-466	#11-479			
L10002	002220		13-489	#13-505			
L10003	012572		#86-2513				
L10004	013660		#96-2717				
L10005	013714		#96-2730				
L10006	015004		#101-2945				
L10007	016064		#103-3082				
L10010	016262		#105-3113				
L10011	016356		#107-3162				
L10012	016422		#109-3200				
L10013	016504		#111-3245				
L10014	016700		114-3312	#114-3319			
L10015	017144		116-3326	#116-3346			
L10016	020534		118-3356	#118-3388			
L10017	021510		120-3396	#120-3420			
L10020	022274		122-3427	#122-3451			
L10021	023250		124-3458	#124-3482			
L10022	024034		126-3489	#126-3513			
L10023	036052		132-3633	#132-3648			
L10024	036242		134-3677	#134-3693			
L10025	036564		#136-3717				
L10027	036574		136-3717	#136-3722			
MABEE	012300		84-2461	#84-2467			
MSAUTO	016320		105-3121	#105-3126			
MSBDA	002332	G	#22-717	98-2788			
MSCMD	002676	G	20-701	#22-739			
MSIOM	002376	G	20-691	#22-721			
MSG1	036052		132-3636	#132-3650			

SYMBOL	VALUE		REFERENCES								
MSG1B	036063		132-3637	#132-3651							
MSG1C	036100		132-3638	#132-3652							
MSG2	036131		132-3639	#132-3653							
MSG3	036146		132-3640	#132-3654							
MSG4	036242		134-3679	#134-3695							
MSG4B	036307		134-3680	#134-3696							
MSG5	036351		134-3681	#134-3697							
MSG6	036403		134-3682	#134-3698							
MSG7	036430		134-3683	#134-3699							
MSG8	036456		134-3685	#134-3700							
MSHCHK	002550	G	20-694	#22-731							
MSHDRD	003146	G	20-692	#22-755							
MSHDWR	003210	G	20-693	#22-757							
MSNIT	002612	G	20-698	#22-733							
MSNLOG	002314	G	20-685	20-689	#22-715						
MSNOMO	002440	G	20-697	#22-723							
MSNOTP	002456	G	20-707	#22-725							
MSNRSP	002756	G	20-705	#22-745							
MSOVRN	003252	G	20-690	#22-759							
MSPART	002626	G	20-699	#22-735							
MSQRSP	002772	G	20-706	#22-747							
MSREC	002712	G	20-702	#22-741							
MSRNIT	002530	G	20-688	#22-729							
MSSELF	002356	G	20-703	#22-719							
MSSFRD	003046	G	20-686	#22-751							
MSSFWR	003106	G	20-687	#22-753							
MSSKER	002300	G	20-695	#22-713							
MSTOSN	003024	G	20-708	#22-749							
MSUNIT	002650	G	20-700	#22-737							
MSWPRO	002506	G	20-696	#22-727							
MSWRSP	002732	G	20-704	#22-743							
MXRTRY	003322	G	#24-783	80-2278							
NCART	= 000054	G	#16-608	82-2373							
NODRVS	016116		103-3015	#103-3087							
NOMOR	006340		60-1650	#60-1657							
NOMOT	= 000030	G	#16-598	82-2350							
NOUNIT	= 000036	G	#16-601	82-2383							
NOXOFF	006430		#62-1701								
NTSFT	012250		84-2454	#84-2460							
NXTRET	006334		60-1620	60-1649	#60-1654						
NXTST	006052	G	58-1582	#60-1611							
NXTST2	006156		60-1622	#60-1626							
ONEFIL	= 000001		#2-4	2-8	4-356	5-357	5-396	13-508	14-509	14-522	100-2844
			101-2845	101-2858	111-3247	11[illegible]3248	1[illegible]2-3259	131-3579	132-3580	132-3594	
OTL	= 000052	G	#16-607	78-2160							
OVRFLO	012746		84-2437	#86-2518							
OVRN	= 000012	G	#16-591	78-2163							
O$APTS	= 000000		#5-375	5-418							
O$AU	= 000001		#5-375	#5-408	5-418						
O$BGNR	= 000001		#5-375	#5-408	5-418						
O$BGNS	= 000001		#5-375	#5-408	5-418						
O$DU	= 000001		#5-375	#5-408	5-418						

SYMBOL	VALUE		REFERENCES								
O$ERRT	= 000000		#5-375	5-418							
O$GNSW	= 000001		#5-375	#5-408	5-418						
O$POIN	= 000001		#5-375	#5-408	#5-408	#5-408	#5-408	#5-408	#5-408	5-408	5-418
O$SETU	= 000001		#5-375	#5-408	5-418	136-3713					
PARTL	= 000034	G	#16-600	82-2393							
PATTEN	= 000072	G	#26-841	86-2505	98-2779	*103-3040	*118-3362	118-3363	*120-3402	*120-3405	120-3406
			*122-3433	*122-3436	*124-3464	*124-3467	124-3468	*126-3495	*126-3498		
PDTFLG	016152	G	*103-3009	103-3036	#103-3090						
PERDEV	006170		#60-1628	60-1647							
PKPTR	= 000104	G	#26-847	*62-1712	70-1908	70-1947	*70-1949	*78-2133	98-2771	100-2820	
PNT	= 001000	G	#14-567								
PRBUF	002210		#13-493	100-2818							
PRDAT	014344		*100-2822	100-2823	#100-2836						
PRFORM	014346		100-2823	#100-2837							
PRI	= 002000	G	#14-567								
PR100	= 000000	G	#14-567	94-2663							
PR101	= 000040	G	#14-567								
PR102	= 000100	G	#14-567								
PR103	= 000140	G	#14-567								
PR104	= 000200	G	#14-567								
PR105	= 000240	G	#14-567								
PR106	= 000300	G	#14-567								
PR107	= 000340	G	#14-567	94-2664	94-2666	105-3101					
PRNPAK	014176	G	98-2792	#100-2813							
PRNSIZ	003330	G	#24-786	*98-2791	*100-2824						
PTR	016454		*109-3201	109-3202	*109-3205	#109-3208					
RCBCNT	003310		#24-778	*68-1832	*68-1843	*68-1848	*68-1852	*70-1904	*70-1921	*70-1925	*70-1927
			*70-1938	*78-2135	78-2142	78-2204					
RCBFSZ	= 001036	G	#18-645	54-1543	130-3542	130-3546	130-3547	130-3548	130-3549	130-3550	130-3551
			130-3552	130-3553							
RCDB	= 000024	G	#26-827	74-2027	74-2052	94-2662	96-2724	*103-3032			
RCFLG	003306	G	#24-777	*68-1833	68-1839	*70-1902	70-1916	*78-2136			
RCINIT	= 000006	G	#16-590	78-2172							
RCSR	= 000022	G	#26-826	74-2025	74-2041	94-2686	94-2703	96-2723	*103-3030	105-3106	
RCVBUF	= 000102	G	#26-846	54-1542	62-1712	68-1829	78-2127	*103-2985			
RCVHND	013662		#96-2723								
RCVINT	013662	G	94-2664	#96-2721							
RDN0	= 000114	G	#26-851	*62-1730	101-2922						
RDN1	= 000116	G	#26-852	*62-1732	101-2932						
REC	= 000064	G	#26-837	62-1708	86-2504	*103-3042	*116-3330	116-3332	*118-3361	118-3363	118-3363
			118-3364	118-3364	*118-3370	118-3371	*118-3374	*118-3375	*120-3399	120-3402	120-3406
			120-3406	*120-3409	*120-3414	*122-3430	122-3433	122-3437	122-3437	*122-3440	*122-3445
			*124-3461	124-3464	124-3468	124-3468	*124-3471	*124-3476	*126-3492	126-3495	126-3499
			126-3499	*126-3502	*126-3507						
RECDAT	017130		116-3330	116-3335	#116-3340						
RECERR	= 000042	G	#16-603	82-2398							
RECID	012654	G	86-2506	#86-2516							
RECID2	013030		86-2509	#86-2520							
RECOV	011262		80-2255	#80-2306							
RETERR	011464		80-2274	#80-2316							
RETRY	= 000002	G	#26-818	*80-2249	80-2250	80-2255	*80-2267	*80-2275	80-2278	*80-2280	80-2281
			*80-2296	*103-3041							

SYMBOL	VALUE		REFERENCES									
RLUN	015006		*101-2911	*101-2912	101-2921	#101-2946						
RPTR	015010		*101-2902	101-2907	101-2933	*101-2935	#101-2947					
RSCMND	= 000002	G	#18-627	18-633	62-1718	114-3314	116-3332	116-3332	118-3363	118-3363	118-3364	
			118-3364	120-3406	120-3406	122-3437	122-3437	124-3468	124-3468	126-3499	126-3499	
RSCONT	= 000020	G	#18-628	78-2139	94-2691	118-3363	118-3363	120-3406	120-3406	124-3468	124-3468	
RSDASZ	= 000204	G	#18-640	18-642	18-645	68-1848	78-2177	78-2184				
RSDATA	= 000001	G	#18-632	68-1846	70-1923	70-1947	78-2154	78-2175	118-3363	118-3363	118-3364	
			118-3364	120-3406	120-3406	122-3437	122-3437	124-3468	124-3468	126-3499	126-3499	
RSDNSZ	= 000222	G	#18-642	70-1925								
RSEND	= 000002	G	#18-633	68-1841	70-1919	78-2149	78-2187	114-3314	118-3363	118-3363	118-3364	
			118-3364	120-3406	120-3406	122-3437	122-3437	124-3468	124-3468	126-3499	126-3499	
RSINIT	= 000004	G	#18-631	78-2169	96-2747							
RSMSIZ	= 000012	G	#18-638	18-644	114-3314	114-3314	116-3332	116-3332	118-3363	118-3363	118-3363	
			118-3363	118-3364	118-3364	118-3364	118-3364	120-3406	120-3406	120-3406	120-3406	
			122-3437	122-3437	122-3437	122-3437	124-3468	124-3468	124-3468	124-3468	126-3499	
			126-3499	126-3499	126-3499							
RSNDSZ	= 000016	G	#18-636	18-642	18-645	68-1843	70-1921	114-3314	116-3332	118-3363	118-3363	
			118-3364	118-3364	120-3406	120-3406	122-3437	122-3437	124-3468	124-3468	126-3499	
			126-3499									
RSNTAB	002220		#20-685	84-2442								
RSSEND	= 000100	G	#18-651									
RSSNIT	= 000001	G	#18-656	94-2645								
RSSNOP	= 000000	G	#18-655									
RSSNSZ	= 000016	G	#18-644	78-2190	114-3314	116-3332	118-3363	118-3363	118-3364	118-3364	120-3406	
			120-3406	122-3437	122-3437	124-3468	124-3468	126-3499	126-3499			
RSSRD	= 000002	G	#18-653	62-1725	82-2341	118-3363	118-3364	118-3364	120-3406	122-3437	122-3437	
			124-3468	126-3499	126-3499							
RSSSEK	= 000005	G	#18-654	116-333c								
RSSSLF	= 000007	G	#18-657	82-2358	114-3314							
RSSWR	= 000003	G	#18-652	62-1734	82-2345	118-3363	120-3406	124-3468				
RSVP	006364	G	#62-1c93	114-3314	116-3332	118-3363	118-3363	118-3363	118-3364	1 8-3364	120-3406	
			120-3406	120-3406	122-3437	122-3437	124-3468	124-3468	124-3468	126-3499	126-3499	
RSXOFF	= 000023	G	#18-63(	68-1870	130-3539	130-3540						
RSXON	= 000020	G	#18-62?	68-1869								
RTRYN	011422		80-2250	80-2281	#80-2312							
RUN	006022	G	#58-1582	58-1591	114-3311	114-3311	116-3325	116-3325	118-3355	118-3355	120-3395	
			120-3395	122-3426	122-3426	124-3457	124-3457	126-3488	126-3488			
SECREC	003326	G	#24-785	*103-3056	*103-3059	120-3414	122-3445	124-3476	126-3507			
SERVST	007656		*66-1795	70-1957	*72-1982	#72-1996						
SETDR	005602	G	#50-1497	114-3311	116-3325	118-3355	120-3395	122-3426	124-3457	126-3488		
SETLEN	016026		#103-3054									
SETPTR	005656		*50-1497	50-1498	50-1503	*50-1505	#50-1508					
SETSRV	007572		70-1895	70-1899	#72-1972							
SETUP	005754	G	#56-1564	11,-3311	114-331'	16-3325	116-3325	118-3355	118-3355	120-3395	120-3395	
			122-3426	1. 2-3426	124-34,,	124-3457	126-3488	126-3488				
SFPTBL	002204	G	#13-489									
SFT	012316		84-2470	#84-2473								
SFTOUT	036242		#134-36?3									
SFTRD	= 000002	G	#16-588	80-2263								
SFTWR	= 000004	G	#16-589	80-2265								
SKERR	= 000024	G	#16-596	82-2368								
SLFLW	= 000044	G	#16-604	82-2363								

```
SYMBOL    VALUE          REFERENCES
SND        006434          62-1700    #62-1702    62-1706
SNDBYT     006666   G      62-1702    #64-1763    68-1827     70-1907     70-1951     74-2040     74-2048     74-2060
SNDCNT   = 000070   G     #26-840     *62-1699   *62-1705    *114-3314   *116-3332   *118-3363   *118-3363   *118-3363   *118-3363
                          *118-3364  *118-3364   *120-3406   *120-3406   *120-3406   *120-3406   *122-3437   *122-3437   *124-3468
                          *124-3468  *124-3468   *124-3468   *126-3499   *126-3499
SNDHND     013646         #96-2715
SNDINT     013646   G      94-2666    #96-2713
SOFTR    = 000122   G     #28-864     101-2913    101-2923
SOFTW    = 000124   G     #28-865     101-2915    101-2925
SRVTBL     007636          72-1976    #72-1987
STAEOP     002206         #13-492     107-3139
STATHD     015012          101-2903   #101-2949
STATUS   = 000000   G     #26-817
STHD2      015266          101-2905   #101-2958
STRT       016150   G     *103-2974  *103-2977    103-3018   #103-3089
SUCCS    = 000076   G     #26-843    *62-1710     80-2237     80-2253    *82-2335     86-2506    *103-3043
SUCOTL   = 000046   G     #16-605     82-2401
SVCGBL   = 000000         #5-375     #5-384       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-420
                           5-420      5-420       7-428       7-428       7-428       9-448       9-448       9-448       11-466
                           11-466     11-466      11-466      11-466      11-466      13-489      13-489      13-489      13-489
                           13-489     13-489      32-946      32-946      32-946      86-2496     86-2496     86-2496     96-2713
                           96-2713    96-2713     96-2721     96-2721     96-2721     101-2893    101-2893    101-2893    103-2971
                           103-2971   103-2971    105-3099    105-3099    105-3099    107-3136    107-3136    107-3136    109-3172
                           109-3172   109-3172    111-3222    111-3222    111-3222    132-3633    132-3633    132-3633    134-3677
                           134-3677   134-3677   #136-3713    136-3713    136-3713    136-3713
SVCINS   = 177777         #5-375     #5-381       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-418       5-418       5-418       5-418       5-418
                           5-418      5-418       5-418       5-418       5-420       5-420       9-448       9-448       9-448
                           9-448      9-448       9-448       9-448       9-448       11-466      13-489      32-946      32-946
                           58-1588    60-1650     60-1650     60-1650     60-1650     60-1651     60-1653     64-1768     74-2046
                           80-2250    80-2250     80-2250     80-2250     80-2250     80-2250     80-2255     80-2255     80-2255
                           80-2255    80-2255     80-2255     80-2258     80-2258     80-2258     80-2258     80-2258     80-2260
                           80-2260    80-2260     80-2260     80-2260     80-2274     80-2274     80-2274     80-2274     80-2274
                           80-2281    80-2281     80-2281     80-2281     80-2281     80-2281     80-2289     80-2289     80-2289
                           80-2289    80-2289     84-2437     84-2437     84-2437     84-2437     84-2451     84-2451     84-2451
                           84-2451    84-2457     84-2457     84-2457     84-2457     84-2464     84-2464     84-2464     84-2464
                           84-2475    84-2475     84-2475     84-2475     84-2479     84-2479     84-2479     84-2479     84-2483
```

SYMBOL	VALUE		REFERENCES								
			86-2503	86-2503	86-2503	86-2503	86-2503	86-2503	86-2503	86-2503	86-2503
			86-2503	86-2506	86-2506	86-2506	86-2506	86-2506	86-2506	86-2506	86-2506
			86-2506	86-2506	86-2506	86-2506	86-2509	86-2509	86-2509	86-2509	86-2509
			86-2509	86-2513	94-2648	94-2663	94-2663	94-2664	94-2664	94-2664	94-2664
			94-2664	94-2664	94-2666	94-2666	94-2666	94-2666	94-2666	94-2666	94-2704
			94-2704	94-2706	94-270	96-2717	96-2730	96-2738	96-2739	98-2788	98-2788
			98-2788	98-2788	98-278	98-2789	98-2789	98-2789	98-2789	98-2789	100-2823
			100-2823	100-2823	100-2823	100-2823	100-2823	100-2823	100-2828	100-2828	100-2828
			100-2828	100-2828	100-2830	100-2830	100-2830	100-2830	100-2830	101-2901	101-2903
			101-2903	101-2903	101-2903	101-2903	101-2904	101-2905	101-2905	101-2905	101-2905
			101-2905	101-2906	101-2921	101-2921	101-2921	101-2921	101-2921	101-2921	101-2922
			101-2922	101-2922	101-2922	101-2922	101-2922	101-2922	101-2922	101-2922	101-2922
			101-2922	101-2922	101-2922	101-2922	101-2932	101-2932	101-2932	101-2932	101-2932
			101-2932	101-2932	101-2932	101-2932	101-2932	101-2932	101-2932	101-2932	101-2932
			101-2945	103-2975	103-2975	103-2976	103-2995	103-2995	103-2995	103-2995	103-2996
			103-3002	103-3002	103-3002	103-3003	103-3015	103-3015	103-3015	103-3015	103-3016
			103-3052	103-3052	103-3082	105-3101	105-3101	105-3101	105-3101	105-3101	105-3101
			-3112	105-3112	105-3113	105-3121	105-3121	105-3121	105-3121	105-3121	105-3124
			3141	107-3162	109-3180	109-3180	109-3180	109-3180	109-3180	109-3180	109-3200
			111-3245	114-3312	114-3312	114-3319	116-3326	116-3326	116-3346	118-3356	118-3356
			118-3388	120-3396	120-3396	120-3420	122-3427	122-3427	122-3451	124-3458	124-3458
			124-3482	126-3489	126-3489	126-3513	132-3633	132-3636	132-3636	132-3636	132-3636
			132-3637	132-3637	132-3637	132-3637	132-3638	132-3638	132-3638	132-3639	132-3639
			132-3639	132-3640	132-3640	132-3640	132-3648	134-3677	134-3679	134-3679	134-3679
			134-3679	134-3679	134-3680	134-3680	134-3680	134-3681	134-3681	134-3681	134-3682
			134-3682	134-3682	134-3683	134-3683	134-3683	134-3685	134-3685	134-3685	134-3685
			134-3685	134-3693	136-3713	136-3713	136-3713	136-3717	136-3717		
SVCSUB	= 177777		#5-375	#5-383							
SVCTAG	= 177777		#5-375	#5-385	11-479	11-479	11-479	13-505	13-505	13-505	86-2513
			86-2513	86-2513	96-2717	96-2717	96-2717	96-2730	96-2730	96-2730	101-2945
			101-2945	101-2945	103-3082	103-3082	103-3082	105-3113	105-3113	105-3113	107-3162
			107-3162	107-3162	109-3200	109-3200	109-3200	111-3245	111-3245	111-3245	114-3319
			114-3319	114-3319	116-3346	116-3346	116-3346	118-3388	118-3388	118-3388	120-3420
			120-3420	120-3420	122-3451	122-3451	122-3451	124-3482	124-3482	124-3482	126-3513
			126-3513	126-3513	132-3648	132-3648	132-3648	134-3693	134-3693	134-3693	136-3717
			136-3717	136-3717	136-3722	136-3722	136-3722				
SVCTST	= 177777		#5-375	#5-382	114-3310	114-3310	114-3310	116-3324	116-3324	116-3324	118-3354
			118-3354	118-3354	120-3394	120-3394	120-3394	122-3425	122-3425	122-3425	124-3456
			124-3456	124-3456	126-3487	126-3487	126-3487				
SWAPDR	005500	G	#48-1462	114-3311	116-3325	118-3355	120-3395	122-3426	124-3457	126-3488	
SWPTR	005600		•48-1463	48-1464	48-1473	•48-1475	#48-1483				
SYSTAT	003300	G	#24-766	•60-1617	66-1793	•68-1838	•70-1915	76-2087	•78-2134	80-2230	•80-2299
			86-2503	•90-2555	•90-2558	90-2569	•94-2697	•103-305			
S$LSYM	= 010000		#5-375	#11-479	#13-505	#86-2513	#96-2717	#96-2730	#101-2945	#103-3082	#105-3113
			#107-3162	#109-3200	#111-3245	#114-3319	#116-3346	#118-3388	#120-3420	#122-3451	#124-3482
			#126-3513	#132-3648	#134-3693						
TAPLEN	003302	G	#24-775	•103-3054	•103-3055	103-3057	120-3400	120-3415	122-3431	122-3446	124-3462
			124-3477	126-3493	126-3508						
TMRSHI	011570		80-2260	#80-2310							
TMRSLO	011542		80-2258	#80-2308							
TMP	000066	G	#26-839	•118-3359	118-3375	•118-3376	•120-3400	•120-3407	•120-3415	•122-3431	•122-3438
			•122-3446	•124-3462	•124-3469	•124-3477	•126-3493	•126-3500	•126-3508		

SYMBOL	VALUE		REFERENCES								
TOMANY	016066		103-2995	#103-3085							
TORCVB	= 000050	G	#16-606	74-2066	96-2742						
TOSNDB	= 000056	G	#16-609	64-1773	94-2654						
TRBUF	024370		62-1698	62-1701	62-1707	114-3314	116-3332	118-3363	118-3363	118-3363	118-3363
			118-3364	118-3364	120-3406	120-3406	120-3406	120-3406	122-3437	122-3437	124-3468
			124-3468	124-3468	124-3468	126-3499	126-3499	#130-3542			
TRK	= 000062	G	#26-836	*120-3401	120-3411	*120-3413	*122-3432	122-3442	*122-3444	*124-3463	124-3473
			*124-3475	*126-3494	126-3504	*126-3506					
TRPHND	016266		105-3101	#105-3121							
TRPPTR	016264		*105-3102	105-3103	105-3108	*105-3110	#105-3114				
TSTPC	000020	C	#26-825	*56-1567	60-1619	60-1640	*62-1694				
TSTTOP	003320		#24-782	56-1567	*114-3311	*116-3325	*118-3355	*120-3395	*122-3426	*124-3457	*126-3488
TST1	016552		114-3311	#114-3314							
TST2	016746		116-3325	#116-3329							
TST3	017212		118-3355	#118-3359							
TST3PT	020522		118-3362	118-3366	#118-3381						
TST4	020602		120-3395	#120-3399							
TST4EX	021502		120-3412	#120-3417							
TST5	021556		122-3426	#122-3430							
TST5EX	022266		122-3443	#122-3448							
TST6	022342		124-3457	#124-3461							
TST6EX	023242		124-3474	#124-3479							
TST7	023316		126-3488	#126-3492							
TST7EX	024026		126-3505	#126-3510							
TUVECT	= 000204	G	#28-894	94-2664	*94-2665	94-2666	*94-2667	94-2704	*94-2705	94-2706	*94-2707
			*103-3006								
TSARGC	= 000002		#5-418	5-418	#5-418	5-418	5-418	#5-418	5-418	5-418	#5-418
			5-418	5-418	#5-418	5-418	5-418	#5-418	5-418	5-418	#80-2250
			80-2250	#80-2250	80-2250	80-2250	#80-2255	80-2255	#80-2255	80-2255	80-2255
			#80-2258	80-2258	80-2258	#80-2260	80-2260	80-2260	#80-2274	80-2274	80-2274
			#80-2281	80-2281	#80-2281	80-2281	80-2281	#80-2289	80-2289	80-2289	#86-2503
			86-2503	#86-2503	86-2503	#86-2503	86-2503	#86-2503	86-2503	86-2503	#86-2506
			86-2506	#86-2506	86-2506	#86-2506	86-2506	#86-2506	86-2506	#86-2506	86-2506
			86-2506	#86-2509	86-2509	#86-2509	86-2509	86-2509	#98-2789	98-2789	#98-2789
			98-2789	98-2789	#100-2823	100-2823	#100-2823	100-2823	100-2823	#100-2828	100-2828
			100-2828	#100-2830	100-2830	100-2830	#101-2903	101-2903	101-2903	#101-2905	101-2905
			101-2905	#101-2921	101-2921	#101-2921	101-2921	101-292	#101-2922	101-2922	#101-2922
			101-2922	#101-2922	101-2922	#101-2922	101-2922	#101-2922	101-2922	#101-2922	101-2922
			#101-2922	101-2922	#101-2922	101-2922	#101-2922	101-2922	101-2922	#101-2932	101-2932
			#101-2932	101-2932	#101-2932	101-2932	#101-2932	101-2932	#101-2932	101-2932	#101-2932
			101-2932	#101-2932	101-2932	#101-2932	101-2932	#101-2932	101-2932	101-2932	#105-3121
			105-3121	105-3121	#109-3180	109-3180	#109-3180	109-3180	109-3180		
TSCODE	= 005052		#132-3636	132-3636	#132-3636	132-3636	#132-3636	132-3636	#132-3637	132-3637	#132-3637
			132-3637	#132-3637	132-3637	#132-3638	132-3638	#132-3638	132-3638	#132-3638	132-3638
			#132-3639	132-3639	#132-3639	132-3639	#132-3639	132-3639	#132-3640	132-3640	#132-3640
			132-3640	#132-3640	132-3640	#134-3679	134-3679	#134-3679	134-3679	#134-3679	134-3679
			#134-3680	134-3680	#134-3680	134-3680	#134-3680	134-3680	*134-3681	134-3681	#134-3681
			134-3681	#134-3681	134-3681	#134-3682	134-3682	#134-3682	134-3682	#134-3682	134-3682
			#134-3683	134-3683	#134-3683	134-3683	#134-3683	134-3683	#134-3685	134-3685	#134-3685
			134-3685	#134-3685	134-3685						
TSFRAM	= 000146		#5-375	#60-1650	60-1650	#84-2437	84-2437	#84-2451	84-2451	#84-2457	84-2457
			#84-2464	84-2464	#84-2475	84-2475	#84-2479	84-2479	#98-2788	98-2788	#103-2995

SYMBOL	VALUE	REFERENCES								
		103-2995	#103-3015	103-3015						
T$EXCP	= 000000	#132-3636	132-3636	#132-3637	132-3637	#134-3679	134-3679	#134-3685	134-3685	
T$FLAG	= 000040	#114-3312	#114-3312	114-3312	114-3312	#116-3326	#116-3326	116-3326	116-3326	#118-3356
		#118-3356	118-3356	118-3356	#120-3396	#120-3396	120-3396	120-3396	#122-3427	#122-3427
		122-3427	122-3427	#124-3458	#124-3458	124-3458	124-3458	#126-3489	#126-3489	126-3489
		126-3489								
T$FREE	= 036574	136-3713	#136-3727							
T$GMAN	= 000000	#5-375								
T$HILI	= 000376	#132-3636	132-3636	#132-3637	132-3637	#134-3679	134-3679	#134-3685	134-3685	
T$LAST	= 000001	#5-375	#136-3713	136-3716						
T$LOLI	= 000001	#132-3636	132-3636	#132-3637	132-3637	#134-3679	134-3679	#134-3685	134-3685	
T$LSYM	= 010000	#5-375	5-375	11-479	13-505	86-2513	96-2717	96-2730	101-2945	103-3082
		105-3113	107-3162	109-3200	111-3245	114-3319	116-3346	118-3388	120-3420	122-3451
		124-3482	126-3513	132-3648	134-3693					
T$LTNO	= 000007	#136-3713								
T$NEST	= 177777	#5-375	5-401	#5-401	5-401	7-428	#7-428	7-428	7-432	7-432
		7-432	#7-432	11-466	#11-466	11-466	11-479	11-479	11-479	#11-479
		13-489	#13-489	13-489	13-505	13-505	13-505	#13-505	13-507	13-507
		13-507	#13-507	14-560	#14-560	14-560	86-2496	#86-2496	86-2496	86-2513
		86-2513	86-2513	#86-2513	96-2713	#96-2713	96-2713	96-2717	96-2717	96-2717
		#96-2717	96-2721	#96-2721	96-2721	96-2730	96-2730	96-2730	#96-2730	100-2842
		100-2842	100-2842	#100-2842	101-2886	#101-2886	101-2886	101-2893	#101-2893	101-2893
		101-2945	101-2945	101-2945	#101-2945	101-2961	101-2961	101-2961	#101-2961	103-2971
		#103-2971	103-2971	103-3082	103-3082	103-3082	#103-3082	105-3099	#105-3099	105-3099
		105-3113	105-3113	105-3113	#105-3113	107-3136	#107-3136	107-3136	107-3162	107-3162
		107-3162	#107-3162	109-3172	#109-3172	109-3172	109-3200	109-3200	109-3200	#109-3200
		111-3222	#111-3222	111-3222	111-3245	111-3245	111-3245	#111-3245	114-3307	#114-3307
		114-3307	114-3310	#114-3310	114-3310	114-3319	114-3319	114-3319	#114-3319	116-3324
		#116-3324	116-3324	116-3346	116-3346	116-3346	#116-3346	118-3354	#118-3354	118-3354
		118-3388	118-3388	118-3388	#118-3388	120-3394	#120-3394	120-3394	120-3420	120-3420
		120-3420	#120-3420	122-3425	#122-3425	122-3425	122-3451	122-3451	122-3451	#122-3451
		124-3456	#124-3456	124-3456	124-3482	124-3482	124-3482	#124-3482	126-3487	#126-3487
		126-3487	126-3513	126-3513	126-3513	#126-3513	130-3557	130-3557	130-3557	#130-3557
		132-3622	#132-3622	132-3622	132-3633	#132-3633	132-3633	132-3648	132-3648	132-3648
		#132-3648	134-3677	#134-3677	134-3677	134-3693	134-3693	134-3693	#134-3693	136-3714
		136-3714	136-3714	#136-3714						
T$NSO	= 000000	#5-401	13-507	#14-560	100-2842	#101-2886	101-2961	#103-2971	103-3082	#105-3099
		105-3113	#107-3136	107-3162	#109-3172	109-3200	#111-3222	111-3245	#114-3307	130-3557
		#132-3622	136-3714							
T$[illegible]	= 000005	#7-428	7-432	#11-466	11-479	#13-489	13-505	#86-2496	86-2513	#96-2713
		96-2717	#96-2721	96-2730	#101-2893	101-2945	#114-3310	114-3319	#116-3324	116-3346
		#118-3354	118-3388	#120-3394	120-3420	#122-3425	122-3451	#124-3456	124-3482	#126-3487
		126-3513	#132-3633	132-3648	#134-3677	134-3693				
T$PCNT	= 000000	#136-3716	136-3717	#136-3717	136-3717					
T$PTAB	= 010026	#136-3717	136-3717							
T$PTMV	= 000001	5-418	#136-3723							
T$PTNU	= 000001	#5-375	136-3717	#136-3717	136-3723	136-3723				
T$SAVL	= 177777	#5-375								
T$SEGL	= 177777	#5-375								
T$SIZE	000006	136-3713	#136-3723							
T$SUBN	= 000000	#5-375	#114-3310	#116-3324	#118-3354	#120-3394	#122-3425	#124-3456	#126-3487	
T$TAGL	= 177777	#5-375								

CZTUUB CREATED BY MACRO ON 2-AUG-79 AT 15:57 PAGE 16
SYMBOL CROSS REFERENCE CREF V01 SEQ 0109

SYMBOL	VALUE	REFERENCES								
T$TAGN	= 010030	#5-375	7-428	7-428	#7-428	11-466	11-466	#11-466	13-489	13-489
		#13-489	86-2496	86-2496	#86-2496	96-2713	96-2713	#96-2713	96-2721	96-2721
		#96-2721	101-2893	101-2893	#101-2893	103-2971	103-2971	#103-2971	105-3099	105-3099
		#105-3099	107-3136	107-3136	#107-3136	109-3172	109-3172	#109-3172	111-3222	111-3222
		#111-3222	114-3310	114-3310	#114-3310	116-3324	116-3324	#116-3324	118-3354	118-3354
		#118-3354	120-3394	120-3394	#120-3394	122-3425	122-3425	#122-3425	124-3456	124-3456
		#124-3456	126-3487	126-3487	#126-3487	132-3633	132-3633	#132-3633	134-3677	134-3677
		#134-3677	136-3716	136-3716	#136-3716	136-3717	136-3717	#136-3717	136-3717	136-3717
		#136-3717								
T$TEMP	= 000000	#7-432	7-432	#9-448	9-448	9-448	#9-448	9-448	9-448	#9-448
		9-448	9-448	#9-448	9-448	9-448	#9-448	9-448	9-448	#9-448
		9-448	9-448	#9-448	9-448	9-448	#9-448	#11-479	11-479	#13-505
		13-505	#13-507	13-507	#86-2513	86-2513	#96-2717	96-2717	#96-2730	96-2730
		#100-2842	100-2842	#101-2945	101-2945	#101-2961	101-2961	#103-3082	103-3082	#105-3113
		105-3113	#107-3162	107-3162	#109-3200	109-3200	#111-3245	111-3245	#114-3312	114-3312
		#114-3319	114-3319	#116-3326	116-3326	#116-3346	116-3346	#118-3356	118-3356	#118-3388
		118-3388	#120-3396	120-3396	#120-3420	120-3420	#122-3427	122-3427	#122-3451	122-3451
		#124-3458	124-3458	#124-3482	124-3482	#126-3489	126-3489	#126-3513	126-3513	#130-3557
		130-3557	#132-3636	132-3636	#132-3636	132-3636	#132-3636	132-3636	#132-3637	132-3637
		#132-3637	132-3637	#132-3637	132-3637	#132-3638	132-3638	#132-3638	132-3638	#132-3638
		132-3638	#132-3639	132-3639	#132-3639	132-3639	#132-3639	132-3639	#132-3640	132-3640
		#132-3640	132-3640	#132-3640	132-3640	#132-3648	132-3648	#134-3679	134-3679	#134-3679
		134-3679	#134-3679	134-3679	#134-3680	134-3680	#134-3680	134-3680	#134-3680	134-3680
		#134-3681	134-3681	#134-3681	134-3681	#134-3681	134-3681	#134-3682	134-3682	#134-3682
		134-3682	#134-3682	134-3682	#134-3683	134-3683	#134-3683	134-3683	#134-3683	134-3683
		#134-3685	134-3685	#134-3685	134-3685	#134-3685	134-3685	#134-3693	134-3693	#136-3714
		136-3714								
T$TEST	= 000007	#5-375	114-3310	#114-3310	114-3310	116-3324	#116-3324	116-3324	118-3354	#118-3354
		118-3354	120-3394	#120-3394	120-3394	122-3425	#122-3425	122-3425	124-3456	#124-3456
		124-3456	126-3487	#126-3487	126-3487	136-3713				
T$TSTM	= 177777	#5-375	58-1588	60-1650	60-1651	60-1653	64-1768	74-2046	80-2250	80-2255
		80-2258	80-2260	80-2274	80-2281	80-2289	84-2437	84-2451	84-2457	84-2464
		84-2475	84-2479	84-2483	86-2503	86-2506	86-2509	86-2513	94-2648	94-2663
		94-2664	94-2666	94-2704	94-2706	96-2738	96-2739	98-2788	98-2789	100-2823
		100-2828	100-2830	101-2901	101-2903	101-2904	101-2905	101-2906	101-2921	101-2922
		101-2932	101-2945	103-2975	103-2995	103-2996	103-3002	103-3015	103-3016	103-3052
		103-3082	105-3101	105-3112	105-3113	105-3121	105-3124	107-3141	107-3162	109-3180
		109-3200	111-3245	114-3312	114-3319	116-3326	116-3346	118-3356	118-3388	120-3396
		120-3420	122-3427	122-3451	124-3458	124-3482	126-3489	126-3513		
T$TSTS	= 000001	#5-375	#114-3310	#116-3324	#118-3354	#120-3394	#122-3425	#124-3456	#126-3487	
T$$AU	= 010013	#111-3222	111-3245							
T$$AUT	= 010010	#105-3099	105-3113							
T$$CLE	= 010011	#107-3136	107-3162							
T$$DAT	= 010027	#136-3717	136-3717	136-3722						
T$$DU	= 010012	#109-3172	109-3200							
T$$HAR	= 010023	#132-3633	132-3633	132-3648						
T$$HW	= 010001	#11-466	11-466	11-479						
T$$INI	= 010007	#103-2971	103-3082							
T$$MSG	= 010003	#86-2496	86-2513							
T$$PC	= 000001	#136-3716	136-3723							
T$$PRO	= 010000	#7-428								
T$$PTA	= 010026	#136-3716	136-3717	#136-3717						

```
SYMBOL  VALUE           REFERENCES
T$$RPT  = 010006        #101-2893   101-2945
T$$SOF  = 010024        #134-3677   134-3677   134-3693
T$$SRV  = 010005        #96-2713    96-2717    #96-2721    96-2730
T$$SW   = 010002        #13-489     13-489     13-505
T$$TES  = 010022        #114-3310   114-3312   114-3319   #116-3324   116-3326   116-3346   #118-3354   118-3356   118-3388
                        #120-3394   120-3396   120-3420   #122-3425   122-3427   122-3451   #124-3456   124-3458   124-3482
                        #126-3487   126-3489   126-3513
T1      016506  G       9-448      #114-3310
T1TRY   = 000146  G     #28-877
T2      016702  G       9-448      #116-3324
T3      017146  G       9-448      #118-3354
T4      020536  G       9-448      #120-3394
T4TRY   = 000132  G     #28-871
T5      021512  G       9-448      #122-3425
T6      022276  G       9-448      #124-3456
T7      023252  G       9-448      #126-3487
UAM     = 000200  G     #14-567
UNIT    012574  G       86-2503    #86-2514
UNREC   011442          80-2289    #80-2314
UNSUC   011040          80-2254    #80-2271
UNXPCT  007356          #70-1918
WAIT    013716          94-2672     94-2680    94-2687    #96-2734    96-2741
WHCMDR  013052  G       62-1728     62-1742    84-2432    #88-2532
WRLOCK  = 000026  G     #16-597     82-2388    84-2460
WRTN0   = 000110  G     #26-849    *62-1744    101-2922    103-3022   103-3024
WRTN1   = 000112  G     #26-850    *62-1747    101-2932
XFNSND  006416          #62-1698
XMDB    = 000030  G     #26-829     64-1776    94-2657    96-2716    *103-3039
XMSR    = 000026  G     #26-828     64-1745    94-2644    94-2650    94-2661    94-2671    94-2679    94-2702    96-2715
                        *103-3034
XSCNT   = 000036  G     #26-832    *114-3314  *116-3332  *118-3363  *118-3363  *118-3363  *120-3406  *120-3406  *120-3406
                       *124-3468  *124-3468  *124-3468
XSFLG   = 000034  G     #26-831     62-1713    68-1822    78-2129   *114-3314  *116-3332  *118-3363  *118-3363  *118-3363
                        118-3363   118-3364   118-3364  *120-3406  *120-3406  *120-3406   120-3406   122-3437   122-3437
                       *124-3468  *124-3468  *124-3468   124-3468   126-3499   126-3499
XSPKNM  = 000032  G     #26-830     62-1709   *62-1709    68-1830    70-1897   *70-1901   *70-1912   *70-1918   *70-1934
                       *70-1945     78-2128  *114-3314  *116-3332  *118-3363  *118-3363  *118-3363  *118-3363  *118-3364
                       *118-3364  *120-3406  *120-3406  *120-3406  *120-3406  *122-3437  *122-3437  *124-3468  *124-3468
                       *124-3468  *124-3468  *126-3499  *126-3499
XSPTR   = 000106  G     #26-848    *62-1715    70-1902   *70-1903    70-1904   *70-1952
XSALWA  = 000000        #5-375
XSFALS  = 000040        #5-375
XSOFFS  = 000400        #5-375
XSTRUE  = 000020        #5-375
```

MACRO NAME	REFERENCES									
BGNAU	#111-3222									
BGNAUT	105-3099									
BGNCLN	#107-3136									
BGNDU	#109-3172									
BGNHRD	132-3633									
BGNHW	#11-466									
BGNINI	103-2971									
BGNMOD	5-401	14-560	101-2886	114-3307	132-3622					
BGNMSG	#86-2496									
BGNPRO	7-428									
BGNPTA	#136-3717									
BGNRPT	101-2893									
BGNSET	#136-3716									
BGNSFT	#134-3677									
BGNSRV	96-2713	96-2721								
BGNSW	#13-489									
BGNTST	114-3310	116-3324	118-3354	120-3394	122-3425	124-3456	126-3487			
BNCOMP	103-2976	103-3003								
BREAK	58-1588	60-1651	64-1768	74-2046	94-2648	96-2738	96-2739	101-2901	101-2904	101-2906
CLRVEC	94-2704	94-2706	105-3112							
DESCRI	#5-420									
DEVTYP	#32-946									
DISPAT	9-448									
DOCLN	#60-1653	#103-2996	#103-3016							
DODU	84-2483	105-3124								
DORPT	107-3141									
ENDAU	111-3245									
ENDAUT	#105-3113									
ENDCLN	107-3162									
ENDDU	109-3200									
ENDHRD	#132-3648									
ENDHW	11-479									
ENDINI	#103-3082									
ENDMOD	13-507	100-2842	101-2961	130-3557	136-3714					
ENDMSG	86-2513									
ENDPRO	#7-432									
ENDPTA	136-3722									
ENDRPT	#101-2945									
ENDSET	#136-3723									
ENDSFT	134-3693									
ENDSRV	#96-2712	#96-2730								
ENDSW	13-505									
ENDTST	114-3314	116-3346	118-3388	120-3420	122-3451	124-3482	126-3513			
EQUALS	14-567									
ERRDF	84-2437	84-2451								
ERRHRD	#84-2464	#84-2479								
ERRSF	#60-1650	#103-2995	#103-3015							
ERRSOF	84-2457	84-2475	98-2788							
EXIT	#114-3312	#116-3326	#118-3356	#120-3396	#122-3427	#124-3458	#126-3489			
GPHARD	103-3002									
GPRMA	#132-3636	#132-3637								
GPRMD	134-3679	134-3685								

```
MACRO NAME  REFERENCES
GPRML       #132-3638  #132-3639  #132-3640  #134-3680  #134-3681  #134-3682  #134-3683
HEADER       5-418
LASTAD       136-3713
MSBYTE      #5-418     #5-418     #5-418     #5-418
MSCHEC      #114-3312   114-3312  #116-3326   116-3326  #118-3356   118-3356  #120-3396   120-3396  #122-3427   122-3427
            #124-3458   124-3458  #126-3489   126-3489
MSCNTO      #132-3636   132-3636  #132-3637   132-3637  #132-3638   132-3638  #132-3639   132-3639  #132-3640   132-3640
            #134-3679   134-3679  #134-3680   134-3680  #134-3681   134-3681  #134-3682   134-3682  #134-3683   134-3683
            #134-3685   134-3685
MSCOUN      #80-2250    80-2250   #80-2255    80-2255   #80-2258    80-2258   #80-2260    80-2260   #80-2274    80-2274
            #80-2281    80-2281   #80-2289    80-2289   #86-2503    86-2503    86-2503    86-2503   #86-2506    86-2506
             86-2506    86-2506    86-2506   #86-2509    86-2509   #98-2789    98-2789   #100-2823   100-2823  #100-2828
             100-2828  #100-2830   100-2830  #101-2903   101-2903  #101-2905   101-2905  #101-2921   101-2921  #101-2922
             101-2922   101-2922   101-2922   101-2922   101-2922   101-2922   101-2922   101-2922  #101-2932   101-2932
             101-2932   101-2932   101-2932   101-2932   101-2932   101-2932   101-2932  #105-3121   105-3121  #109-3180
             109-3180
MSDATA      #5-418      5-418      5-418      5-418      5-418      5-418      5-418      5-418      5-418      5-418
             5-418      5-418      5-418      5-418      5-418      5-418     #5-418      5-418      5-418      5-418
             5-418      5-418      5-418      5-418      5-418      5-418      5-418      5-418      5-418      5-418
             5-418      5-418      5-418      5-418      5-418      5-418      5-418      5-418      5-418     #5-420
             5-420     #32-946     32-946
MSDECR      #7-432      7-432     #11-479     11-479    #13-505     13-505    #13-507     13-507    #86-2513    86-2513
            #96-2717    96-2717   #96-2730    96-2730   #100-2842   100-2842  #101-2945   101-2945  #101-2961   101-2961
            #103-3082   103-3082  #105-3113   105-3113  #107-3162   107-3162  #109-3200   109-3200  #111-3245   111-3245
            #114-3319   114-3319  #116-3346   116-3346  #118-3388   118-3388  #120-3420   120-3420  #122-3451   122-3451
            #124-3482   124-3482  #126-3513   126-3513  #130-3557   130-3557  #132-3648   132-3648  #134-3693   134-3693
            #136-3714   136-3714  #136-3717   136-3717
MSDEFA      #132-3636  #132-3636  #132-3637  #132-3637  #132-3638  #132-3638  #132-3639  #132-3639  #132-3640  #132-3640
            #134-3679  #134-3679  #134-3680  #134-3680  #134-3681  #134-3681  #134-3682  #134-3682  #134-3683  #134-3683
            #134-3685  #134-3685
MSENDE      #11-479    #13-505    #13-507    #86-2513   #96-2717   #96-2730   #100-2842  #101-2945  #101-2961  #103-3082
            #105-3113  #107-3162  #109-3200  #111-3245  #114-3319  #116-3346  #118-3388  #120-3420  #122-3451  #124-3482
            #126-3513  #130-3557  #132-3648  #134-3693  #136-3714
MSERRI      #60-1650   #60-1650   #84-2437   #84-2437   #84-2451   #84-2451   #84-2457   #84-2457   #84-2464   #84-2464
            #84-2475   #84-2475   #84-2479   #84-2479   #98-2788   #98-2788   #103-2995  #103-2995  #103-3015  #103-3015
MSEXCP      #132-3636   132-3636   132-3636  #132-3637   132-3637   132-3637  #134-3679   134-3679   134-3679  #134-3685
             134-3685   134-3685
MSEXIT      #114-3312   114-3312  #116-3326   116-3326  #118-3356   118-3356  #120-3396   120-3396  #122-3427   122-3427
            #124-3458   124-3458  #126-3489   126-3489
MSEXSE      #114-3312  #116-3326  #118-3356  #120-3396  #122-3427  #124-3458  #126-3489
MSEXTJ      #114-3312  #116-3326  #118-3356  #120-3396  #122-3427  #124-3458  #126-3489
MSGEN       #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
            #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
            #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
            #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
            #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
            #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
            #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
            #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-420     #5-420     #7-428     #7-428
            #9-448     #9-448     #11-466    #11-466    #11-466    #11-466    #11-479    #11-479    #13-489    #13-489
            #13-489    #13-489    #13-505    #13-505    #32-946    #32-946    #86-2496   #86-2496   #86-2513   #86-2513
            #96-2713   #96-2713   #96-2717   #96-2717   #96-2721   #96-2721   #96-2730   #96-2730   #101-2893  #101-2893
```

```
MACRO NAME   REFERENCES
             #101-2945  #101-2945  #103-2971  #103-2971  #103-3082  #103-3082  #105-3099  #105-3099  #105-3113  #105-3113
             #107-3136  #107-3136  #107-3162  #107-3162  #109-3172  #109-3172  #109-3200  #109-3200  #111-3222  #111-3222
             #111-3245  #111-3245  #114-3310  #114-3310  #114-3319  #114-3319  #116-3324  #116-3324  #116-3346  #116-3346
             #118-3354  #118-3354  #118-3388  #118-3388  #120-3394  #120-3394  #120-3420  #120-3420  #122-3425  #122-3425
             #122-3451  #122-3451  #124-3456  #124-3456  #124-3482  #124-3482  #126-3487  #126-3487  #126-3513  #126-3513
             #132-3633  #132-3633  #132-3648  #132-3648  #134-3677  #134-3677  #134-3693  #134-3693  #136-3713  #136-3713
             #136-3717  #136-3717  #136-3722  #136-3722
MSGETS       #7-432      7-432     #11-479     11-479    #13-505     13-505    #13-507     13-507    #86-2513    86-2513
             #96-2717    96-2717   #96-2730    96-2730   #100-2842   100-2842  #101-2945   101-2945  #101-2961   101-2961
             #103-3082   103-3082  #105-3113   105-3113  #107-3162   107-3162  #109-3200   109-3200  #111-3245   111-3245
             #114-3319   114-3319  #116-3346   116-3346  #118-3388   118-3388  #120-3420   120-3420  #122-3451   122-3451
             #124-3482   124-3482  #126-3513   126-3513  #130-3557   130-3557  #132-3648   132-3648  #134-3693   134-3693
             #136-3714   136-3714
MSGETI       #114-3312  #116-3326  #118-3356  #120-3396  #122-3427  #124-3458  #126-3489
MSGNGB       #5-401     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
             #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
             #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
             #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
             #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
             #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
             #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418
             #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-418     #5-420     #5-420     #7-428
             #7-428     #9-448     #9-448     #11-466    #11-466    #11-466    #13-489    #13-489    #13-489    #14-560
             #32-946    #32-946    #86-2496   #86-2496   #96-2713   #96-2713   #96-2721   #96-2721   #101-2886  #101-2893
             #101-2893  #103-2971  #103-2971  #105-3099  #105-3099  #107-3136  #107-3136  #109-3172  #109-3172  #111-3222
             #111-3222  #114-3307  #132-3622  #132-3633  #132-3633  #134-3677  #134-3677  #136-3713  #136-3713
MSGNIN       #5-418      5-418      5-418      5-418      5-418      5-418     #5-418      5-418     #5-418      5-418
             #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418
             #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418
             #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418
             #5-418      5-418     #5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418      5-418
             #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418
             #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418
             #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418
             #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418     #5-418      5-418
             #5-420     #5-420      5-420      5-420     #9-448      9-448     #9-448      9-448     #9-448      9-448
             #9-448      9-448     #9-448      9-448     #9-448      9-448     #9-448      9-448     #9-448      9-448
             #11-466     11-466    #13-489     13-489    #32-946    #32-946     32-946     32-946    #58-1588    58-1588
             #60-1650   #60-1650    60-1650   #60-1650    60-1650   #60-1650    60-1650   #60-1650    60-1650   #60-1651
              60-1651   #60-1653    60-1653   #64-1768    64-1768   #74-2046    74-2046   #80-2250   #80-2250    80-2250
             #80-2250    80-2250   #80-2250    80-2250    80-2250   #80-2250    80-2250    80-2250   #80-2255   #80-2255
              80-2255   #80-2255    80-2255   #80-2255    80-2255    80-2255   #80-2255    80-2255    80-2255   #80-2258
             #80-2258    80-2258   #80-2258    80-2258    80-2258   #80-2258    80-2258    80-2258   #80-2260   #80-2260
              80-2260   #80-2260    80-2260    80-2260   #80-2260    80-2260    80-2260   #80-2274   #80-2274    80-2274
             #80-2274    80-2274    80-2274   #80-2274    80-2274    80-2274   #80-2281   #80-2281    80-2281   #80-2281
              80-2281   #80-2281    80-2281    80-2281   #80-2281    80-2281    80-2281   #80-2289   #80-2289    80-2289
             #80-2289    80-2289    80-2289   #80-2289    80-2289    80-2289   #84-2437   #84-2437    84-2437   #84-2437
              84-2437   #84-2437    84-2437   #84-2437    84-2437   #84-2451   #84-2451    84-2451   #84-2451    84-2451
             #84-2451    84-2451   #84-2451    84-2451   #84-2457   #84-2457    84-2457   #84-2457    84-2457   #84-2457
              84-2457   #84-2457    84-2457   #84-2464    84-2464    84-2464   #84-2464    84-2464   #84-2464    84-2464
             #84-2464    84-2464   #84-2475   #84-2475    84-2475   #84-2475    84-2475   #84-2475    84-2475   #84-2475
              84-2475   #84-2479   #84-2479    84-2479   #84-2479    84-2479   #84-2479    84-2479   #84-2479    84-2479
```

MACRO NAME	REFERENCES									
	#84-2483	#84-2483	84-2483	#86-2503	#86-2503	86-2503	86-2503	#86-2503	86-2503	#86-2503
	86-2503	86-2503	#86-2503	86-2503	#86-2503	86-2503	86-2503	#86-2503	86-2503	86-2503
	#86-2506	#86-2506	86-2506	86-2506	#86-2506	86-2506	86-2506	#86-2506	86-2506	86-2506
	#86-2506	86-2506	#86-2506	86-2506	#86-2506	86-2506	86-2506	#86-2506	86-2506	86-2506
	#86-2509	#86-2509	86-2509	#86-2509	86-2509	#86-2509	86-2509	86-2509	#86-2509	86-2509
	86-2509	#86-2513	86-2513	#94-2648	94-2648	#94-2663	94-2663	#94-2663	94-2663	#94-2664
	#94-2664	94-2664	#94-2664	94-2664	#94-2664	94-2664	#94-2664	94-2664	#94-2664	94-2664
	94-2664	#94-2666	#94-2666	94-2666	#94-2666	94-2666	#94-2666	94-2666	#94-2666	94-2666
	#94-2666	94-2666	94-2666	#94-2704	94-2704	#94-2704	94-2704	#94-2706	94-2706	#94-2706
	94-2706	#96-2717	96-2717	#96-2730	96-2730	#96-2738	96-2738	#96-2739	96-2739	#98-2788
	#98-2788	98-2788	#98-2788	98-2788	#98-2788	98-2788	#98-2788	98-2788	#98-2789	#98-2789
	98-2789	#98-2789	98-2789	#98-2789	98-2789	98-2789	#98-2789	98-2789	98-2789	#100-2823
	#100-282?	100-2823	100-2823	#100-2823	100-2823	#100-2823	100-2823	100-2823	#100-2823	100-2823
	100-2823	#100-2828	#100-2828	100-2828	#100-2828	100-2828	100-2828	#100-2828	100-2828	100-2828
	#100-2830	#100-2830	100-2830	#100-2830	100-2830	100-2830	#100-2830	100-2830	100-2830	#101-2901
	101-2901	#101-2903	#101-2903	101-2903	#101-2903	101-2903	101-2903	#101-2903	101-2903	101-2903
	#101-2904	101-2904	#101-2905	#101-2905	101-2905	#101-2905	101-2905	101-2905	#101-2905	101-2905
	101-2905	#101-2906	101-2906	#101-2921	#101-2921	101-2921	#101-2921	101-2921	#101-2921	101-2921
	101-2921	#101-2921	101-2921	101-2921	#101-2922	#101-2922	101-2922	#101-2922	101-2922	#101-2922
	101-2922	#101-2922	101-2922	#101-2922	101-2922	101-2922	#101-2922	101-2922	#101-2922	101-2922
	#101-2922	101-2922	#101-2922	101-2922	#101-2922	101-2922	101-2922	#101-2922	101-2922	101-2922
	#101-2932	#101-2932	101-2932	#101-2932	101-2932	#101-2932	101-2932	#101-2932	101-2932	#101-2932
	101-2932	101-2932	#101-2932	101-2932	#101-2932	101-2932	#101-2932	101-2932	#101-2932	101-2932
	#101-2932	101-2932	101-2932	#101-2932	101-2932	101-2932	#101-2945	101-2945	#103-2975	103-2975
	#103-2975	103-2975	#103-2976	103-2976	#103-2995	#103-2995	103-2995	#103-2995	103-2995	#103-2995
	103-2995	#103-2995	103-2995	#103-2996	103-2996	#103-3002	103-3002	#103-3002	103-3002	#103-3002
	103-3002	#103-3003	103-3003	#103-3015	#103-3015	103-3015	#103-3015	103-3015	#103-3015	103-3015
	#103-3015	103-3015	#103-3016	103-3016	#103-3052	103-3052	#103-3052	103-3052	#103-3082	103-3082
	#105-3101	#105-3101	105-3101	#105-3101	105-3101	#105-3101	105-3101	#105-3101	105-3101	#105-3101
	105-3101	105-3101	#105-3112	105-3112	#105-3112	105-3112	#105-3113	105-3113	#105-3121	#105-3121
	105-3121	#105-3121	105-3121	105-3121	#105-3121	105-3121	105-3121	#105-3124	#105-3124	105-3124
	#107-3141	107-3141	#107-3162	107-3162	#109-3180	#109-3180	109-3180	#109-3180	109-3180	#109-3180
	109-3180	109-3180	#109-3180	109-3180	109-3180	#109-3200	109-3200	#111-3245	111-3245	#114-3312
	114-3312	#114-3312	114-3312	#114-3319	114-3319	#116-3326	116-3326	#116-3326	116-3326	#116-3346
	116-3346	#118-3356	118-3356	#118-3356	118-3356	#118-3388	118-3388	#120-3396	120-3396	#120-3396
	120-3396	#120-3420	120-3420	#122-3427	122-3427	#122-3427	122-3427	#122-3451	122-3451	#124-3458
	124-3458	#124-3458	124-3458	#124-3482	124-3482	#126-3489	126-3489	#126-3489	126-3489	#126-3513
	126-3513	#132-3633	132-3633	#132-3636	132-3636	132-3636	132-3636	132-3636	#132-3637	132-3637
	132-3637	132-3637	132-3637	#132-3638	132-3638	132-3638	132-3638	#132-3639	132-3639	132-3639
	132-3639	#132-3640	132-3640	132-3640	132-3640	#132-3648	132-3648	#134-3677	134-3677	#134-3679
	134-3679	134-3679	134-3679	134-3679	134-3679	#134-3680	134-3680	134-3680	134-3680	#134-3681
	134-3681	134-3681	134-3681	#134-3682	134-3682	134-3682	134-3682	#134-3683	134-3683	134-3683
	134-3683	#134-3685	134-3685	134-3685	134-3685	134-3685	134-3685	#134-3693	134-3693	#136-3713
	136-3713	136-3713	136-3713	#136-3717	#136-3717	136-3717	136-3717			
MSGNTA	#11-479	#11-479	#13-505	#13-505	#86-2513	#86-2513	#96-2717	#96-2717	#96-2730	#96-2730
	#101-2945	#101-2945	#103-3082	#103-3082	#105-3113	#105-3113	#107-3162	#107-3162	#109-3200	#109-3200
	#111-3245	#111-3245	#114-3319	#114-3319	#116-3346	#116-3346	#118-3388	#118-3388	#120-3420	#120-3420
	#122-3451	#122-3451	#124-3482	#124-3482	#126-3513	#126-3513	#132-3648	#132-3648	#134-3693	#134-3693
	#134-3717	#136-3717	#136-3722	#136-3722						
MSGNTE	#114-3310	#114-3310	#116-3324	#116-3324	#118-3354	#118-3354	#120-3394	#120-3394	#122-3425	#122-3425
	#124-3456	#124-3456	#126-3487	#126-3487						
MSMAPT	#5-418	#5-418								

MACRO NAME	REFERENCES									
M$HNAP	#5-418	#5-418								
M$INCR	#5-401	#5-401	#7-428	#7-428	#7-428	#7-428	#11-466	#11-466	#11-466	#11-466
	#13-489	#13-489	#13-489	#13-489	#14-560	#14-560	#58-1588	#60-1650	#60-1651	#60-1653
	#64-1768	#74-2046	#80-2250	#80-2255	#80-2258	#80-2260	#80-2274	#80-2281	#80-2289	#84-2437
	#84-2451	#84-2457	#84-2464	#84-2475	#84-2479	#84-2483	#86-2496	#86-2496	#86-2496	#86-2496
	#86-2503	#86-2506	#86-2509	#86-2513	#94-2648	#94-2663	#94-2664	#94-2666	#94-2704	#94-2706
	#96-2713	#96-2713	#96-2713	#96-2713	#96-2721	#96-2721	#96-2721	#96-2721	#96-2738	#96-2739
	#98-2788	#98-2789	#100-2823	#100-2828	#100-2830	#101-2886	#101-2886	#101-2893	#101-2893	#101-2893
	#101-2893	#101-2901	#101-2903	#101-2904	#101-2905	#101-2906	#101-2921	#101-2922	#101-2932	#101-2945
	#103-2971	#103-2971	#103-2971	#103-2971	#103-2975	#103-2995	#103-2996	#103-3002	#103-3015	#103-3016
	#103-3052	#103-3082	#105-3099	#105-3099	#105-3099	#105-3099	#105-3101	#105-3112	#105-3113	#105-3121
	#105-3124	#107-3136	#107-3136	#107-3136	#107-3136	#107-3141	#107-3162	#109-3172	#109-3172	#109-3172
	#109-3172	#109-3180	#109-3200	#111-3222	#111-3222	#111-3222	#111-3222	#111-3245	#114-3307	#114-3307
	#114-3310	#114-3310	#114-3310	#114-3310	#114-3310	#114-3310	#114-3312	#114-3319	#116-3324	#116-3324
	#116-3324	#116-3324	#116-3324	#116-3324	#116-3326	#116-3346	#118-3354	#118-3354	#118-3354	#118-3354
	#118-3354	#118-3354	#118-3356	#118-3388	#120-3394	#120-3394	#120-3394	#120-3394	#120-3394	#120-3394
	#120-3396	#120-3420	#122-3425	#122-3425	#122-3425	#122-3425	#122-3425	#122-3425	#122-3427	#122-3451
	#124-3456	#124-3456	#124-3456	#124-3456	#124-3456	#124-3456	#124-3458	#124-3482	#126-3487	#126-3487
	#126-3487	#126-3487	#126-3487	#126-3487	#126-3489	#126-3513	#132-3622	#132-3622	#132-3633	#132-3633
	#132-3633	#132-3633	#134-3677	#134-3677	#134-3677	#134-3677	#136-3716	#136-3716	#136-3717	#136-3717
	#136-3717	#136-3717								
M$LDRO	#84-2483	84-2483	#94-2663	94-2663	#94-2704	94-2704	#94-2706	94-2706	#103-2975	103-2975
	#103-3002	103-3002	#105-3112	105-3112	#105-3124	105-3124				
M$MCHI	#5-375	#5-375								
M$MCLO	#5-375	#5-375								
M$POP	#7-432	7-432	#11-479	11-479	#13-505	13-505	#13-507	13-507	#86-2513	86-2513
	#96-2717	96-2717	#96-2730	96-2730	#100-2842	100-2842	#101-2945	101-2945	#101-2961	101-2961
	#103-3082	103-3082	#105-3113	105-3113	#107-3162	107-3162	#109-3200	109-3200	#111-3245	111-3245
	#114-3319	114-3319	#116-3346	116-3346	#118-3388	118-3388	#120-3420	120-3420	#122-3451	122-3451
	#124-3482	124-3482	#126-3513	126-3513	#130-3557	130-3557	#132-3648	132-3648	#134-3693	134-3693
	#136-3714	136-3714								
M$PRIN	#80-2250	#80-2250	#80-2255	#80-2255	#80-2258	#80-2258	#80-2260	#80-2260	#80-2274	#80-2274
	#80-2281	#80-2281	#80-2289	#80-2289	#86-2503	#86-2503	#86-2506	#86-2506	#86-2509	#86-2509
	#98-2789	#98-2789	#100-2823	#100-2823	#100-2828	#100-2828	#100-2830	#100-2830	#101-2903	#101-2903
	#101-2905	#101-2905	#101-2921	#101-2921	#101-2922	#101-2922	#101-2932	#101-2932	#105-3121	#105-3121
	#109-3180	#109-3180								
M$PUSH	#5-401	#5-401	#7-428	#7-428	#11-466	#11-466	#13-489	#13-489	#14-560	#14-560
	#86-2496	#86-2496	#96-2713	#96-2713	#96-2721	#96-2721	#101-2886	#101-2886	#101-2893	#101-2893
	#103-2971	#103-2971	#105-3099	#105-3099	#107-3136	#107-3136	#109-3172	#109-3172	#111-3222	#111-3222
	#114-3307	#114-3307	#114-3310	#114-3310	#116-3324	#116-3324	#118-3354	#118-3354	#120-3394	#120-3394
	#122-3425	#122-3425	#124-3456	#124-3456	#126-3487	#126-3487	#132-3622	#132-3622	#132-3633	#132-3633
	#134-3677	#134-3677								
M$PUT	#80-2250	#80-2250	#80-2250	#80-2250	#80-2255	#80-2255	#80-2255	#80-2255	#80-2258	#80-2258
	#80-2258	#80-2260	#80-2260	#80-2260	#80-2274	#80-2274	#80-2274	#80-2281	#80-2281	#80-2281
	#80-2281	#80-2289	#80-2289	#80-2289	#86-2503	#86-2503	#86-2503	#86-2503	#86-2503	#86-2503
	#86-2506	#86-2506	#86-2506	#86-2506	#86-2506	#86-2506	#86-2506	#86-2509	#86-2509	#86-2509
	#86-2509	#94-2664	#94-2664	#94-2664	#94-2664	#94-2664	#94-2666	#94-2666	#94-2666	#94-2666
	#94-2666	#98-2789	#98-2789	#98-2789	#98-2789	#100-2823	#100-2823	#100-2823	#100-2823	#100-2828
	#100-2828	#100-2828	#100-2830	#100-2830	#100-2830	#101-2903	#101-2903	#101-2903	#101-2905	#101-2905
	#101-2905	#101-2921	#101-2921	#101-2921	#101-2921	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922
	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922	#101-2932	#101-2932	#101-2932	#101-2932
	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#105-3101	#105-3101	#105-3101

MACRO NAME	REFERENCES									
	#105-3101	#105-3101	#105-3121	#105-3121	#105-3121	#109-3180	#109-3180	#109-3180	#109-3180	
MSPUT1	#80-2250	#80-2250	#80-2250	#80-2250	#80-2250	#80-2250	#80-2255	#80-2255	#80-2255	#80-2255
	#80-2255	#80-2255	#80-2258	#80-2258	#80-2258	#80-2258	#80-2260	#80-2260	#80-2260	#80-2260
	#80-2274	#80-2274	#80-2274	#80-2274	#80-2281	#80-2281	#80-2281	#80-2281	#80-2281	#80-2281
	#80-2289	#80-2289	#80-2289	#80-2289	#86-2503	#86-2503	#86-2503	#86-2503	#86-2503	#86-2503
	#86-2503	#86-2503	#86-2503	#86-2503	#86-2506	#86-2506	#86-2506	#86-2506	#86-2506	#86-2506
	#86-2506	#86-2506	#86-2506	#86-2506	#86-2506	#86-2506	#86-2509	#86-2509	#86-2509	#86-2509
	#86-2509	#86-2509	#94-2664	#94-2664	#94-2664	#94-2664	#94-2664	#94-2664	#94-2664	#94-2664
	#94-2666	#94-2666	#94-2666	#94-2666	#94-2666	#94-2666	#94-2666	#94-2666	#98-2789	#98-2789
	#98-2789	#98-2789	#98-2789	#98-2789	#100-2823	#100-2823	#100-2823	#100-2823	#100-2823	#100-2823
	#100-2828	#100-2828	#100-2828	#100-2828	#100-2830	#100-2830	#100-2830	#100-2830	#101-2903	#101-2903
	#101-2903	#101-2903	#101-2905	#101-2905	#101-2905	#101-2905	#101-2921	#101-2921	#101-2921	#101-2921
	#101-2921	#101-2921	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922
	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922
	#101-2922	#101-2922	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932
	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932
	#101-2932	#101-2932	#105-3101	#105-3101	#105-3101	#105-3101	#105-3101	#105-3101	#105-3101	#105-3101
	#105-3121	#105-3121	#105-3121	#105-3121	#109-3180	#109-3180	#109-3180	#109-3180	#109-3180	#109-3180
MSRADI	#132-3636	#132-3636	#132-3637	#132-3637	#132-3638	#132-3638	#132-3639	#132-3639	#132-3640	#132-3640
	#134-3679	#134-3679	#134-3680	#134-3680	#134-3681	#134-3681	#134-3682	#134-3682	#134-3683	#134-3683
	#134-3685	#134-3685								
MSRNRO	#103-3002	#103-3002	#103-3052	#103-3052						
MSSETS	#5-401	#5-401	#7-428	#7-428	#11-466	#11-466	#13-489	#13-489	#14-560	#14-560
	#86-2496	#86-2496	#96-2713	#96-2713	#96-2721	#96-2721	#101-2886	#101-2886	#101-2893	#101-2893
	#103-2971	#103-2971	#105-3099	#105-3099	#107-3136	#107-3136	#109-3172	#109-3172	#111-3222	#111-3222
	#114-3307	#114-3307	#114-3310	#114-3310	#116-3324	#116-3324	#118-3354	#118-3354	#120-3394	#120-3394
	#122-3425	#122-3425	#124-3456	#124-3456	#126-3487	#126-3487	#132-3622	#132-3622	#132-3633	#132-3633
	#134-3677	#134-3677								
MSSVC	#58-1588	58-1588	60-1650	#60-1651	60-1651	#60-1653	60-1653	#64-1768	64-1768	#74-2046
	74-2046	#80-2250	80-2250	#80-2255	80-2255	#80-2258	80-2258	#80-2260	80-2260	#80-2274
	80-2274	#80-2281	80-2281	#80-2289	80-2289	84-2437	84-2451	84-2457	84-2464	84-2475
	84-2479	#84-2483	84-2483	#86-2503	86-2503	#86-2506	86-2506	#86-2509	86-2509	#86-2513
	86-2513	#94-2648	94-2648	#94-2663	94-2663	#94-2664	94-2664	#94-2666	94-2666	#94-2704
	94-2704	#94-2706	94-2706	#96-2738	96-2738	#96-2739	96-2739	98-2788	#98-2789	98-2789
	#100-2823	100-2823	#100-2828	100-2828	#100-2830	100-2830	#101-2901	101-2901	#101-2903	101-2903
	#101-2904	101-2904	#101-2905	101-2905	#101-2906	101-2906	#101-2921	101-2921	#101-2922	101-2922
	#101-2932	101-2932	#101-2945	101-2945	#103-2975	103-2975	103-2995	#103-2996	103-2996	#103-3002
	103-3002	103-3015	#103-3016	103-3016	#103-3052	103-3052	#103-3082	103-3082	#105-3101	105-3101
	#105-3112	105-3112	#105-3113	105-3113	#105-3121	105-3121	#105-3124	105-3124	#107-3141	107-3141
	#107-3162	107-3162	#109-3180	109-3180	#109-3200	109-3200	#111-3245	111-3245	#114-3312	114-3312
	#114-3319	114-3319	#116-3326	116-3326	#116-3346	116-3346	#118-3356	118-3356	#118-3388	118-3388
	#120-3396	120-3396	#120-3420	120-3420	#122-3427	122-3427	#122-3451	122-3451	#124-3458	124-3458
	#124-3482	124-3482	#126-3489	126-3489	#126-3513	126-3513				
MSTLAB	#58-1588	#60-1650	#60-1651	#60-1653	#64-1768	#74-2046	#80-2250	#80-2255	#80-2258	#80-2260
	#80-2274	#80-2281	#80-2289	#84-2437	#84-2451	#84-2457	#84-2464	#84-2475	#84-2479	#84-2483
	#86-2503	#86-2506	#86-2509	#86-2513	#94-2648	#94-2663	#94-2664	#94-2666	#94-2704	#94-2706
	#96-2738	#96-2739	#98-2788	#98-2789	#100-2823	#100-2828	#100-2830	#101-2901	#101-2903	#101-2904
	#101-2905	#101-2906	#101-2921	#101-2922	#101-2932	#101-2945	#103-2975	#103-2995	#103-2996	#103-3002
	#103-3015	#103-3016	#103-3052	#103-3082	#105-3101	#105-3112	#105-3113	#105-3121	#105-3124	#107-3141
	#107-3162	#109-3180	#109-3200	#111-3245	#114-3312	#114-3319	#116-3326	#116-3346	#118-3356	#118-3388
	#120-3396	#120-3420	#122-3427	#122-3451	#124-3458	#124-3482	#126-3489	#126-3513		
MSTSTL	#58-1588	58-1588	#60-1650	#60-1650	60-1650	#60-1651	60-1651	#60-1653	60-1653	#64-1768

MACRO NAME	REFERENCES									
	64-1768	#74-2046	74-2046	#80-2250	80-2250	#80-2255	80-2255	#80-2258	80-2258	#80-2260
	80-2260	#80-2274	80-2274	#80-2281	80-2281	#80-2289	80-2289	#84-2437	#84-2437	84-2437
	#84-2451	#84-2451	84-2451	#84-2457	#84-2457	84-2457	#84-2464	#84-2464	84-2464	#84-2475
	#84-2475	84-2475	#84-2479	#84-2479	84-2479	#84-2483	84-2483	#86-2503	86-2503	#86-2506
	86-2506	#86-2509	86-2509	#86-2513	86-2513	#94-2648	94-2648	#94-2663	94-2663	#94-2664
	94-2664	#94-2666	94-2666	#94-2704	94-2704	#94-2706	94-2706	#96-2738	96-2738	#96-2739
	96-2739	#98-2788	#98-2788	98-2788	#98-2789	98-2789	#100-2823	100-2823	#100-2828	100-2828
	#100-2830	100-2830	#101-2901	101-2901	#101-2903	101-2903	#101-2904	101-2904	#101-2905	101-2905
	#101-2906	101-2906	#101-2921	101-2921	#101-2922	101-2922	#101-2932	101-2932	#101-2945	101-2945
	#103-2975	103-2975	#103-2995	#103-2995	103-2995	#103-2996	103-2996	#103-3002	103-3002	#103-3015
	#103-3015	103-3015	#103-3016	103-3016	#103-3052	103-3052	#103-3082	103-3082	#105-3101	105-3101
	#105-3112	105-3112	#105-3113	105-3113	#105-3121	105-3121	#105-3124	105-3124	#107-3141	107-3141
	#107-3162	107-3162	#109-3180	109-3180	#109-3200	109-3200	#111-3245	111-3245	#114-3312	114-3312
	#114-3319	114-3319	#116-3326	116-3326	#116-3346	116-3346	#118-3356	118-3356	#118-3388	118-3388
	#120-3396	120-3396	#120-3420	120-3420	#122-3427	122-3427	#122-3451	122-3451	#124-3458	124-3458
	#124-3482	124-3482	#126-3489	126-3489	#126-3513	126-3513				
MSWORD	#5-418	#5-418	#9-448	#9-448	#9-448	#9-448	#9-448	#9-448	#9-448	#9-448
	#9-448	#60-1650	#60-1650	#60-1650	#60-1650	#84-2437	#84-2437	#84-2437	#84-2437	#84-2451
	#84-2451	#84-2451	#84-2451	#84-2457	#84-2457	#84-2457	#84-2457	#84-2464	#84-2464	#84-2464
	#84-2464	#84-2475	#84-2475	#84-2475	#84-2475	#84-2479	#84-2479	#84-2479	#84-2479	#98-2788
	#98-2788	#98-2788	#98-2788	#103-2995	#103-2995	#103-2995	#103-2995	#103-3015	#103-3015	#103-3015
	#103-3015	#114-3312	#116-3326	#118-3356	#120-3396	#122-3427	#124-3458	#126-3489	#132-3636	#132-3636
	#132-3637	#132-3637	#132-3638	#132-3638	#132-3639	#132-3639	#132-3640	#132-3640	#134-3679	#134-3679
	#134-3680	#134-3680	#134-3681	#134-3681	#134-3682	#134-3682	#134-3683	#134-3683	#134-3685	#134-3685
	#136-3717	#136-3717								
POINTE	5-408									
POP	#34-993	54-1547	54-1548	64-1769	64-1777	72-1983	72-1984	80-2301	80-2302	84-2484
	84-2485	84-2486	84-2487	86-2511	86-2512	90-2577	90-2578	92-2615	92-2616	98-2793
	98-2794	98-2795	100-2831	100-2832	101-2939	101-2940	101-2941	101-2942	101-2943	101-2944
	109-317[illegible]	109-3179								
PRINTB	#80-2274	#86-2503	#86-2506	#86-2509	#98-2789					
PRINTF	100-2823	100-2828	100-2830	105-3121	109-3180					
PRINTS	#101-2903	#101-2905	#101-2921	#101-2922	#101-2932					
PRINTX	80-2250	80-2255	80-2258	80-2260	80-2281	80-2289				
PUSH	#34-989	54-1540	54-1541	64-1763	64-1767	72-1972	72-1973	80-2228	80-2229	84-242[illegible]
	84-2422	84-2423	84-2424	86-2497	86-2498	90-2553	90-2554	92-2593	92-2594	98-276[illegible]
	98-2768	98-2769	100-2816	100-2817	101-2894	101-2895	101-2896	101-2897	101-2898	101-2899
	109-3174	109-3175								
READEF	#103-2975									
RFLAGS	#103-3052									
SETPRI	94-2663									
SETVEC	94-2664	94-2666	105-3101							
SVC	#5-374	5-375								
SWAPIN	#34-1002	60-1618	60-1639							
SWAPOW	#34-101[illegible]	#62-1695								
TSTID	#46-1364	#114-3311	#116-3325	#118-3355	#120-3395	#122-3426	#124-3457	#[illegible]88		
TUREAD	#42-1262	118-3364	122-3437	126-3499						
TURTRY	#40-1195	118-3363	118-3364	120-3406	122-3437	124-3468	126-3499			
TUSEEK	#38-1143	116-3332								
TUSELF	#44-1331	114-3314								
TUWRIT	#36-1048	#118-3363	#120-3406	#124-3468						
XFER	#114-3312	#116-3326	#118-3356	#120-3396	#122-3427	#124-3458	#126-[illegible]89			