```
                        .REM    &

                                    IDENTIFICATION
                                    --------------

                          PRODUCT CODE:       AC-E451A-MC

                          PRODUCT NAME:       CXTSAA0 DEC/X11 TS11/TS04 MOD

                          PRODUCT DATE:       FEB 1979

                          MAINTAINER:         DEC/X11 SUPPORT GROUP

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

                          THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED TO THE
                          PURCHASER UNDER A LICENSE FOR USE ON A SINGLE COMPUTER
                          SYSTEM AND CAN BE COPIED (WITH INCLUSION OF DIGITALS
                          COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT AS MAY
                          OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

                          DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR
                          THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS
                          NOT SUPPLIED BY DIGITAL.

                        COPYRIGHT (C)  1978,1979 DIGITAL EQUIPMENT CORPORATION
```

1.0 ABSTRACT

TSA IS AN IOMODX MODULE THAT CAN EXERCISE UP TO 4 TS11/TS04 MAGNETIC TAPE SUBSYSTEMS. IT EXERCISES THE DRIVES BY DOING A WRITE, READ REVERSE, IN-CORE COMPARE, READ FORWARD, IN CORE COMPARE. THIS SEQUENCE OF FUNCTIONS WILL BE DEFINED AS A CYCLE. AN "END OF PASS" WILL BE REACHED AFTER 512 CYCLES. ALL ERRORS DETECTED ARE REPORTED ON THE CONSOLE TTY.

2.0 REQUIREMENTS

HARDWARE: 1 TO 4 TS11/TS04 TAPE SUBSYSTEMS.

STORAGE: TSA REQUIRES 1345 WORDS OF STORAGE.

3.0 PASS DEFINITION

ONE PASS OF THE TSA MODULE CONSISTS OF 512 CYCLES OF THE BASIC TEST SEQUENCE (WRITE, READ REVERSE, DATA COMPARE, READ FORWARD, DATA COMPARE). THE WRITE AND READ DATA LENGTHS ARE 256 WORDS.

4.0 EXECUTION TIME

ONE PASS OF TSA RUNNING ALONE ON A PDP-11/34 TAKES APPROXIMATELY 1 MINUTE.

5.0 CONFIGURATION REQUIREMENTS

DEFAULT PARAMETERS:

DEVADR: 172520 VECTOR: 224 BR1: 5 DEVCNT: 1

REQUIRED PARAMETERS:

(REQUIRED FOR MULTIPLE DRIVE OPERATION ONLY)

SR2: VECTOR FOR 2ND DRIVE

SR3: VECTOR FOR 3RD DRIVE

SR4: VECTOR FOR 4TH DRIVE

6.0 DEVICE/OPTION SETUP

MAKE CERTAIN THAT ALL DRIVES ARE POWERED UP, WRITE ENABLED AND ON LINE. HAVING ALL DRIVES AT LOAD POINT IS NOT ESSENTIAL TO THE OPERATION OF THE MODULE BUT WILL ENSURE THAT THE RECORD COUNT ACCURATELY REFLECTS THE NUMBER OF RECORDS FROM LOAD POINT ON THE 1ST PASS OF TAPE. (THE RECORD COUNT ACCOMPANIES ALL ERROR PRINTOUTS AND CAN BE USED TO IDENTIFY BAD SPOTS ON TAPE).

7.0 MODULE OPERATION

TEST SEQUENCE:

A. SET UP DEVICE REGISTER ADDRESSES AND MODULE VARIABLES.

B. INITIALIZE ALL DEVICES.

C. GET A DEVICE ADDRESS.

D. ISSUE SET CHARACTERISTIC AND ERASE COMMANDS TO THE DEVICE.

E. IF NOT END OF DEVICES, GO TO C.

F. GET NEXT DEVICE ADDRESS.

G. DO A WRITE - IF ERRORS, REPORT AND RETRY TO THE RETRY LIMIT.

H. DO A READ REVERSE - IF ERRORS, REPORT AND RETRY TO THE RETRY LIMIT.

I. DO A DATA CHECK - IF ERRORS, REPORT AND CONTINUE.

J. DO A READ FORWARD - IF ERRORS, REPORT AND RETRY TO THE RETRY LIMIT.

K. DO A DATA CHECK - IF ERRORS, REPORT AND CONTINUE.

L. IF AT END OF TAPE - DO A REWIND.

M. IF END OF PASS, REPORT AND GO TO C, ELSE GO TO F.

ERROR RECOVERY SEQUENCE:

```
IF RETRY LIMIT HAS BEEN REACHED, THEN:
:  PRINT UNRECOVERABLE ERROR.
:  IF BIT 0 IS SET IN SR1, THEN:
:  :  DROP THIS DEVICE FROM THE TEST SEQUENCE.
ELSE:
:  PRINT RECOVERABLE ERROR.
:  INCREMENT THE RETRY COUNTER.
:  IF THE FUNCTION IS A READ, THEN:
:  :  IF THE RETRY COUNT IS HALF THE LIMIT OR MORE, THEN:
:  :  :  SET THE OPPOSITE DIRECTION BIT IN THE RETRY COMMAND.
:  ISSUE THE RETRY COMMAND TO THE DEVICE.
```

RETRY LIMITS:

FOR READ/WRITE COMMANDS = 16 RETRIES.

FOR ALL OTHER COMMANDS = 8 RETRIES.

```
                8.0     OPERATION OPTIONS
                        -----------------

                        SR1 BIT0 CLEAR (0):

                            IF AN UNRECOVERABLE ERROR OCCURS ON ANY FUNCTION, THE FUNC-
                        TION IS ABORTED AND TESTING CONTINUES.

                        SR1 BIT0 SET (1):

                            IF AN UNRECOVERABLE ERROR OCCURS ON ANY FUNCTION, THE DEVICE
                        IS DROPPED FROM THE TEST CYCLE.

                        SR1 BIT1 CLEAR (0):

                            ALL RECOVERABLE ERRORS ARE REPORTED.

                        SR1 BIT1 SET (1):

                            RECOVERABLE ERRORS ARE NOT REPORTED.

                9.0     NON-STANDARD PRINTOUTS
                        ----------------------

                        ALL PRINTOUTS HAVE THE STANDARD FORMATS DESCRIBED IN THE DEC/X11
                        DOCUMENTS.

                        ERROR MESSAGES DUMP THE CONTENTS OF 10 LOCATIONS IN THE FOLLOWING
                        ORDER:

                        DCNT MTYPE MLEN RFC XSTAT0 XSTAT1 XSTAT2 XSTAT3
                        RECORD RTRYC

                         DCNT   = DEVICE NUMBER (0 - 3).
                         MTYPE  = MESSAGE PACKET 1ST WORD - MESSAGE TYPE.
                         MLEN   = MESSAGE PACKET 2ND WORD - MESSAGE LENGTH.
                         RFC    = MESSAGE PACKET 3RD WORD - RESIDUAL FRAME COUNT.
                         XSTAT0 = MESSAGE PACKET 4TH WORD - EXTENDED STATUS REG 0.
                         XSTAT1 = MESSAGE PACKET 5TH WORD - EXTENDED STATUS REG 1.
                         XSTAT2 = MESSAGE PACKET 6TH WORD - EXTENDED STATUS REG 2.
                         XSTAT3 = MESSAGE PACKET 7TH WORD - EXTENDED STATUS REG 3.
                         RECORD = RECORD COUNT - NUMBER OF RECORDS FROM BOT.
                         RTRYC  = RETRY COUNT (WILL BE 0 ON ORIGINAL ERROR).

                10.0    DEFINITION OF ERRORS
                        --------------------

                        RECOVERABLE ERROR   - TERMINATION CLASS CODE OF 4 OR 5 AND RETRY
                                              LIMIT NOT EXCEEDED.

                        UNRECOVERABLE ERROR - TERMINATION CLASS CODE OF 6 OR TERMINATION
                                              CLASS CODE OF 4 OR 5 AND RETRY LIMIT EXCEEDED.

                &
```

```
                                        IOMODX  <TSAA  >,<172520,224,5,0,0,512.,163,BUFIN,256.,256.
                                        MODULE  150000,TSAA  ,172520,224,5,0,0,512.,163,BUFIN,256.,256.
                                        .TITLE  TSAA DEC/X11 SYSTEM EXERCISER MODULE
                                ;       IOXCOM  VERSION  6       23-MAY-78
                                        .LIST   BIN
                                ;*****************************************************************************
                                BEGIN:
                                MODNAM: .ASCII  /TSAA / ;MODULE NAME.
                                XFLAG:  .BYTE   0PEN            ;USED TO KEEP TRACK OF WBUFF USAGE
                                ADDR:   172520+0                ;1ST DEVICE ADDR.
                                VECTOR: 224+0                   ;1ST DEVICE VECTOR.
                                BR1:    .BYTE   PRTY5+0         ;1ST BR LEVEL.
                                BR2:    .BYTE   PRTY0+0         ;2ND BR LEVEL.
                                DVID1:  0+1                     ;DEVICE INDICATOR 1.
                                SR1:    OPEN                    ;SWITCH REGISTER 1
                                SR2:    OPEN                    ;SWITCH REGISTER 2
                                SR3:    OPEN                    ;SWITCH REGISTER 3
                                SR4:    OPEN                    ;SWITCH REGISTER 4
                                ;*****************************************************************************
                                STAT:   150000                  ;STATUS WORD.
                                INIT:   START                   ;MODULE START ADDR.
                                SPOINT: MODSP                   ;MODULE STACK POINTER.
                                PASCNT: 0                       ;PASS COUNTER.
                                ICONT:  512.                    ;# OF ITERATIONS PER PASS=512.
                                ICOUNT: 0                       ;LOC TO COUNT ITERATIONS
                                SOFCNT: 0                       ;LOC TO SAVE TOTAL SOFT ERRORS
                                HRDCNT: 0                       ;LOC TO SAVE TOTAL HARD ERRORS
                                SOFPAS: 0                       ;LOC TO SAVE SOFT ERRORS PER PASS
                                HRDPAS: 0                       ;LOC TO SAVE HARD ERRORS PER PASS
                                SYSCNT: 0                       ;# OF SYS ERRORS ACCUMULATED
                                RANNUM: 0                       ;HOLDS RANDOM # WHEN RAND MACRO IS CALLED
                                CONFIG:                         ;RESERVED FOR MONITOR USE
                                RES1:   0                       ;RESERVED FOR MONITOR USE
                                RES2:   0                       ;RESERVED FOR MONITOR USE
                                SVR0:   OPEN                    ;LOC TO SAVE R0.
                                SVR1:   OPEN                    ;LOC TO SAVE R1.
                                SVR2:   OPEN                    ;LOC TO SAVE R2.
                                SVR3:   OPEN                    ;LOC TO SAVE R3.
                                SVR4:   OPEN                    ;LOC TO SAVE R4.
                                SVR5:   OPEN                    ;LOC TO SAVE R5.
                                SVR6:   OPEN                    ;LOC TO SAVE R6.
                                CSRA:   OPEN                    ;ADDR OF CURRENT CSR.
                                SBADR:                          ;ADDR OF GOOD DATA, OR
                                ACSR:   OPEN                    ;CONTENTS OF CSR.
                                WASADR:                         ;ADDR OF BAD DATA, OR
                                ASTAT:  OPEN                    ;STATUS REG CONTENTS.
                                ERRTYP:                         ;TYPE OF ERROR
                                ASD:    OPEN                    ;EXPECTED DATA.
                                AWAS:   OPEN                    ;ACTUAL DATA.
                                RSTRT:  RESTRT                  ;RESTART ADDRESS AFTER END OF PASS
                                WDTO:   OPEN                    ;WORDS TO MEMORY PER ITERATION
                                WDFR:   OPEN                    ;WORDS FROM MEMORY PER ITERATION
                                INTR:   OPEN                    ;# OF INTERRUPTS PER ITERATION
                                IDNUM:  163                     ;MODULE IDENTIFICATION NUMBER=163
                                RBUFVA: BUFIN                   ;READ BUFFER VIRTUAL ADDRESS
                                RBUFPA: OPEN                    ;READ BUFFER PHYSICAL ADDRESS
```



```
                                RBUFEA: OPEN                    ;READ BUFFER EA BITS
                                RBUFSZ: 256.                    ;SIZE OF THE READ BUFFER
                                WBUFPA: OPEN                    ;WRITE BUFFER PHYSICAL ADDRESS
                                WBUFEA: OPEN                    ;WRITE BUFFER EA BITS
                                WBUFRQ: 256.                    ;WRITE BUFFER SIZE REQUESTED
                                WBUFSZ: OPEN                    ;WRITE BUFFER SIZE AVAILABLE
                                CDERCT: OPEN                    ;CDATA/DATCK ERROR COUNT
                                CDWDCT: OPEN                    ;CDATA/DATCK WORD COUNT
                                FREE:   OPEN                    ;RESERVED FOR FUTURE USE
                                        .REPT   SPSIZ           ;MODULE STACK STARTS HERE.
                                        .NLIST
                                        .WORD   0
                                        .LIST
                                        .ENDR
                                MODSP:
                                ;*****************************************************************************
```

```
                                        .LIST   MC
                                        .NLIST  ME
                                        .MCALL  STRUCT
                                        STRUCT                          ; INITIALIZE STRUCTURED MACROS
000252'                         .PRINT  ;SPMAC: VERSION 1.1
                000001                  SLSTIM=1                        ; LIST STRUCTURED MACRO INSTRUCTIONS.
                000001                  SLSTTAG=1                       ; LIST STRUCTURED MACRO TAGS.
                                ;       ---------------------------------


                                ;       THE FOLLOWING ARE TS04 COMMAND DEFINITIONS.

                140204                  SCH=140204                      ;SET CHARACTERISTIC.
                100611                  ERS=100611                      ;ERASE.
                100205                  WRT=100205                      ;WRITE.
                100601                  RDR=100601                      ;READ REVERSE.
                100201                  RDF=100201                      ;READ FORWARD.
                102210                  RWD=102210                      ;REWIND.
                101205                  WTR=101205                      ;WRITE RETRY.
                101601                  RRR=101601                      ;READ REVERSE RETRY.
                101201                  RFR=101201                      ;READ FORWARD RETRY.

                                ;       THE FOLLOWING ARE COMMAND WORD BIT DEFINITIONS.

                020000                  CM.OPP=20000                    ;OPPOSITE DIRECTION BIT.

                                ;       THE FOLLOWING ARE TSSR(STATUS REGISTER) BIT DEFINITIONS.

                000200                  TS.SSR=200                      ;SUBSYSTEM READY.
                100000                  TS.SC=100000                    ;SPECIAL CONDITION.

                                ;       THE FOLLOWING ARE EXTENDED STATUS REGISTER 0 BIT DEFINITIONS.

                000001                  XS0.EOT=1                       ;END OF TAPE.

                                ;       MISCELLANEOUS DEFINITIONS.

                000020                  WRETRY=16.                      ;WRITE COMMAND RETRY LIMIT.
                000020                  RRETRY=16.                      ;READ COMMAND RETRY LIMIT.
                000010                  CRETRY=8.                       ;RETRY LIMIT FOR NON-READ/WRITE CMDS.
                000016                  MSGCNT=14.                      ;MESSAGE PACKET LENGTH IN BYTES.
                000004                  SCHCNT=4                        ;SET CHARACTERISTIC BLOCK LENGTH IN WORDS.
                000040                  SCDFLT=40                       ;CHARACTERISTIC CODE DEFAULT.
                177777                  TERM=177777                     ;TABLE TERMINATION INDICATOR.
                                ;       ---------------------------------
```



```
                                ;       THE FOLLOWING LOCATIONS ARE USED TO REQUEST BUFFERS FROM THE MONITOR.

000252' 000622'                 CBUFVA: CMDPKT                          ;COMMAND PACKET VIRTUAL ADDRESS.
000254' 000000                  CBUFPA: 0                               ;COMMAND PACKET PHYSICAL ADDRESS
000256' 000000                  CBUFEA: 0                               ;COMMAND PACKET EA BITS.
000260' 000644'                 MBUFVA: MSGPKT                          ;MESSAGE PACKET VIRTUAL ADDRESS.
000262' 000000                  MBUFPA: 0                               ;MESSAGE PACKET PHYSICAL ADDRESS.
000264' 000000                  MBUFEA: 0                               ;MESSAGE PACKET EA BITS.
000266' 000634'                 SBUFVA: MSGBLK                          ;MESSAGE BLOCK VIRTUAL ADDRESS.
000270' 000000                  SBUFPA: 0                               ;MESSAGE BLOCK PHYSICAL ADDRESS.
000272' 000000                  SBUFEA: 0                               ;MESSAGE BLOCK EA BITS.

                                ;       THE FOLLOWING LOCATIONS ARE THE DEVICE POINTERS.

000274' 000000                  DCNT:   0                               ;DEVICE COUNTER.
000276' 000000                  DINX:   0                               ;DEVICE INDEX.
000300' 000000                  DIND:   0                               ;DEVICE INDICATOR.
000302' 000000                  VDIND:  0                               ;VARIABLE DEVICE INDICATOR.

                                ;       THE FOLLOWING LOCATIONS ARE DEVICE VARIABLES.

000304' 000000                  TSSR:   0                               ;STATUS REGISTER ADDRESS OF CURRENT DEVICE.
000306' 000000                  VECT:   0                               ;VECTOR ADDRESS OF CURRENT DEVICE.
000310' 000000                  RECORD: 0                               ;RECORD COUNT OF CURRENT DEVICE.

                                ;       THE FOLLOWING LOCATIONS ARE COMMAND VARIABLES.

000312' 000000                  CMD:    0                               ;CURRENT TS04 COMMAND WORD.
000314' 000000                  RCMD:   0                               ;RETRY COMMAND WORD.
000316' 000000                  RTRYL:  0                               ;RETRY LIMIT.
000320' 000000                  RTRYC:  0                               ;RETRY COUNT.

                                ;       THE FOLLOWING TABLES ARE USED TO SET THE DEVICE VARIABLES.
                                ;       THESE TABLES ARE INITIALIZED DURING MODULE START INITIALIZATION.

000322' 000000                  TTSDB:  0                               ;TSDB ADR FOR DEVICE 0.
000324' 000000                          0                               ;TSDB ADR FOR DEVICE 1.
000326' 000000                          0                               ;TSDB ADR FOR DEVICE 2.
000330' 000000                          0                               ;TSDB ADR FOR DEVICE 3.
000332' 000000                  TTSSR:  0                               ;TSSR ADR FOR DEVICE 0.
000334' 000000                          0                               ;TSSR ADR FOR DEVICE 1.
000336' 000000                          0                               ;TSSR ADR FOR DEVICE 2.
000340' 000000                          0                               ;TSSR ADR FOR DEVICE 3.
000342' 000000                  TVECT:  0                               ;VECTOR ADR FOR DEVICE 0.
000344' 000000                          0                               ;VECTOR ADR FOR DEVICE 1.
000346' 000000                          0                               ;VECTOR ADR FOR DEVICE 2.
000350' 000000                          0                               ;VECTOR ADR FOR DEVICE 3.
000352' 000000                  TREC:   0                               ;RECORD COUNT FOR DEVICE 0.
000354' 000000                          0                               ;RECORD COUNT FOR DEVICE 1.
000356' 000000                          0                               ;RECORD COUNT FOR DEVICE 2.
000360' 000000                          0                               ;RECORD COUNT FOR DEVICE 3.
```

```
                                        ;       THE FOLLOWING TABLES ARE USED TO SET UP COMMAND VARIABLES.
                                        ;
000362' 140204                  SCHAR:  SCH                             ;SET CHARACTERISTIC COMMAND.
000364' 140204                          SCH                             ;RETRY COMMAND
000366' 000010                          CRETRY                          ;RETRY LIMIT.
000370' 005053'                         SCHAS                           ;ADDRESS OF CMD ASCII.
000372' 000270'                         SBUFPA                          ;ADDRESS OF BUFFER PHYSICAL ADR.
000374' 000272'                         SBUFEA                          ;ADDRESS OF BUFFER EXTENDED ADR.
000376' 000616'                         SBUFSZ                          ;ADDRESS OF BUFFER SIZE.
000400' 100611                  ERASE:  ERS                             ;ERASE COMMAND.
000402' 100611                          ERS                             ;RETRY COMMAND
000404' 000010                          CRETRY                          ;RETRY LIMIT.
000406' 005064'                         ERSAS                           ;ADDRESS OF CMD ASCII.
000410' 000560'                         BCON                            ;ADDRESS OF BUFFER PHYSICAL ADR.
000412' 000560'                         BCON                            ;ADDRESS OF BUFFER EXTENDED ADR.
000414' 000560'                         BCON                            ;ADDRESS OF BUFFER SIZE.
000416' 100205                  WRITE:  WRT                             ;WRITE COMMAND.
000420' 101205                          WTR                             ;RETRY COMMAND
000422' 000020                          WRETRY                          ;RETRY LIMIT.
000424' 005072'                         WRTAS                           ;ADDRESS OF CMD ASCII.
000426' 000134'                         WBUFPA                          ;ADDRESS OF BUFFER PHYSICAL ADR.
000430' 000136'                         WBUFEA                          ;ADDRESS OF BUFFER EXTENDED ADR.
000432' 000142'                         WBUFSZ                          ;ADDRESS OF BUFFER SIZE.
000434' 100601                  RREV:   RDR                             ;READ REVERSE COMMAND.
000436' 101601                          RRR                             ;RETRY COMMAND
000440' 000020                          RRETRY                          ;RETRY LIMIT.
000442' 005100'                         RDRAS                           ;ADDRESS OF CMD ASCII.
000444' 000126'                         RBUFPA                          ;ADDRESS OF BUFFER PHYSICAL ADR.
000446' 000130'                         RBUFEA                          ;ADDRESS OF BUFFER EXTENDED ADR.
000450' 000132'                         RBUFSZ                          ;ADDRESS OF BUFFER SIZE.
000452' 100201                  RFWD:   RDF                             ;READ FORWARD COMMAND.
000454' 101201                          RFR                             ;RETRY COMMAND
000456' 000020                          RRETRY                          ;RETRY LIMIT.
000460' 005111'                         RDFAS                           ;ADDRESS OF CMD ASCII.
000462' 000126'                         RBUFPA                          ;ADDRESS OF BUFFER PHYSICAL ADR.
000464' 000130'                         RBUFEA                          ;ADDRESS OF BUFFER EXTENDED ADR.
000466' 000132'                         RBUFSZ                          ;ADDRESS OF BUFFER SIZE.
000470' 102210                  REWIND: RWD                             ;REWIND COMMAND.
000472' 102210                          RWD                             ;RETRY COMMAND
000474' 000010                          CRETRY                          ;RETRY LIMIT.
000476' 005122'                         RWDAS                           ;ADDRESS OF CMD ASCII.
000500' 000560'                         BCON                            ;ADDRESS OF BUFFER PHYSICAL ADR.
000502' 000560'                         BCON                            ;ADDRESS OF BUFFER EXTENDED ADR.
000504' 000560'                         BCON                            ;ADDRESS OF BUFFER SIZE.

                                        ;       THE FOLLOWING BYTES ARE USED AS PROGRAM CONTROL FLAGS.

000506'     000         EOD:    .BYTE   0                               ;END OF DEVICES.
000507'     000         DROP:   .BYTE   0                               ;CURRENT DEVICE HAS BEEN DROPPED.
000510'     000         DMOD:   .BYTE   0                               ;MODULE IS TO BE DROPPED.
000511'     000         EREC:   .BYTE   0                               ;ERROR RECOVERY IS REQUIRED.
                                .EVEN
```



```
                                        ;       THE FOLLOWING BYTES ARE UNEXPECTED INTERRUPT FLAGS AND MUST BE
                                        ;       ASSEMBLED IN ORDER.

0000512'    000         UIFL0:  .BYTE   0                               ;DEVICE 0.
0000513'    000         UIFL1:  .BYTE   0                               ;DEVICE 1.
0000514'    000         UIFL2:  .BYTE   0                               ;DEVICE 2.
0000515'    000         UIFL3:  .BYTE   0                               ;DEVICE 3.
                                .EVEN

                                        ;       THE FOLLOWING LOCATIONS CONTAIN ADDRESSES OF THE TERMINATION CLASS CODE
                                        ;       HANDLING ROUTINES AND MUST BE ASSEMBLED IN ORDER.

0000516' 003610'                TCCRA:  TCC0                            ;TERMINATION CLASS CODE 0.
0000520' 003660'                        TCC1                            ;TERMINATION CLASS CODE 1.
0000522' 003732'                        TCC2                            ;TERMINATION CLASS CODE 2.
0000524' 004010'                        TCC3                            ;TERMINATION CLASS CODE 3.
0000526' 004062'                        TCC4                            ;TERMINATION CLASS CODE 4.
0000530' 004170'                        TCC5                            ;TERMINATION CLASS CODE 5.
0000532' 004226'                        TCC6                            ;TERMINATION CLASS CODE 6.
0000534' 004234'                        TCC7                            ;TERMINATION CLASS CODE 7.

                                        ;       THE FOLLOWING LOCATIONS CONTAIN ADDRESSES OF UNEXPECTED INTERRUPT
                                        ;       ROUTINES AND MUST BE ASSEMBLED IN ORDER.

0000536' 004454'                UIADR:  UID0                            ;DEVICE 0.
0000540' 004464'                        UID1                            ;DEVICE 1.
0000542' 004474'                        UID2                            ;DEVICE 2.
0000544' 004504'                        UID3                            ;DEVICE 3.

                                        ;       THE FOLLOWING LOCATIONS CONTAIN ADDRESSES OF THE COMMAND PACKET WORDS.
                                        ;       THESE ADDRESSES ARE CALCULATED AND STORED DURING INITIALIZATION.

0000546' 000000                 CMDADR: 0                               ;COMBINED PHYSICAL COMMAND PACKET
                                                                        ;ADDRESS WITH EXTENDED BITS IN BITS 0+1.
0000550' 000000                 CMDPK1: 0                               ;ADDRESS OF COMMAND PACKET 1ST WORD.
0000552' 000000                 CMDPK2: 0                               ;ADDRESS OF COMMAND PACKET 2ND WORD.
0000554' 000000                 CMDPK3: 0                               ;ADDRESS OF COMMAND PACKET 3RD WORD.
0000556' 000000                 CMDPK4: 0                               ;ADDRESS OF COMMAND PACKET 4TH WORD.

                                        ;       THE FOLLOWING CONSTANTS MUST BE ASSEMBLED IN ORDER.

0000560' 000001                 BCON:  1
0000562' 000002                        2
0000564' 000004                        4
0000566' 000010                        10

                                        ;       THE FOLLOWING TABLE CONTAINS ADDRESSES OF VARIABLES TO BE PRINTED
                                        ;       IN THE EXTENDED ERROR REPORTS.

0000570' 000274'                TABLE:  DCNT                            ;DEVICE NUMBER
0000572' 000644'                        MSGPKT                          ;STATUS MESSAGE TYPE.
0000574' 000646'                        MSGPKT+2                        ;STATUS MESSAGE LENGTH IN BYTES.
0000576' 000650'                        MSGPKT+4                        ;RESIDUAL FRAME COUNT.
0000600' 000652'                        MSGPKT+6                        ;EXTENDED STATUS REGISTER 0.
0000602' 000654'                        MSGPKT+10                       ;EXTENDED STATUS REGISTER 1.
0000604' 000656'                        MSGPKT+12                       ;EXTENDED STATUS REGISTER 2.
```

```
000606' 000660'                 MSGPKT+14                       ;EXTENDED STATUS REGISTER 3.
000610' 000310'                 RECORD                          ;RECORD COUNT.
000612' 000320'                 RTRYC                           ;RETRY COUNT.
000614' 177777                  TERM                            ;TABLE TERMINATOR.

                        ;       MISCELLANEOUS PROGRAM VARIABLES.

000616' 000004          SBUFSZ: SCHCNT                          ;SET CHARACTERISTIC BLOCK LENGTH IN WORDS.
000620' 000000          TIME:   0                               ;TIMEOUT COUNTER.

                        ;       THIS IS THE COMMAND PACKET - IT MUST BE ON A MODULO 4 BOUNDRY.

000622' 000000          CMDPKT: 0                               ;TS04 COMMAND WORD.
000624' 000000                  0                               ;BUFFER PHYSICAL ADDRESS.
000626' 000000                  0                               ;BUFFER EXTENDED ADR IN BITS 0+1.
000630' 000000                  0                               ;BUFFER LENGTH IN BYTES.
000632' 000000                  0                               ;EXTENTION TO INSURE MODULE 4 BOUNDRY
                                                                ;(1ST 4 WORDS CAN BE PUSHED DOWN 1 WORD).

                        ;       THIS IS THE SET CHARACTERISTIC BLOCK.

000634' 000000          MSGBLK: 0                               ;MESSAGE PACKET PHYSICAL ADDRESS.
000636' 000000                  0                               ;MESSAGE PACKET EXTENDED ADR IN BITS 0+1.
000640' 000016                  MSGCNT                          ;MESSAGE PACKET LENGTH IN BYTES.
000642' 000040                  SCDFLT                          ;DEFAULT CHARACTERISTIC CODE.

                        ;       THIS IS THE MESSAGE PACKET.

000644' 000000          MSGPKT: 0                               ;MESSAGE TYPE.
000646' 000000                  0                               ;MESSAGE LENGTH IN BYTES.
000650' 000000          RFC:    0                               ;RESIDUAL FRAME COUNT.
000652' 000000          XSTAT0: 0                               ;EXTENDED STATUS REGISTER 0.
000654' 000000          XSTAT1: 0                               ;EXTENDED STATUS REGISTER 1.
000656' 000000          XSTAT2: 0                               ;EXTENDED STATUS REGISTER 2.
000660' 000000          XSTAT3: 0                               ;EXTENDED STATUS REGISTER 3.

                        ;       THIS IS THE READ BUFFER.

000662' 000400          BUFIN:  .BLKW   256.                    ;512 BYTES.
                        ;       ---------------------------------
```



```
001662'                 START:  LET WDTO := RBUFSZ SHIFT 1       ;LET # OF READ WDS PER CYCLE = BUFFER SIZE X2.
001662' 016767  176244  176224                                                   MOV     RBUFSZ,WDTO
001670' 006367  176220                                                          ASL     WDTO
001674'                         LET WDFR := WBUFSZ               ;LET # OF WRITE WDS PER CYCLE = BUFFER SIZE.
001674' 016767  176242  176214                                                  MOV     WBUFSZ,WDFR
001702'                         LET INTR := #3                   ;LET # OF INTERRUPTS PER CYCLE = 3.
001702' 012767  000003  176210                                                  MOV     #3,INTR
001710'                         LET DIND := DVID1                ;LET DEVICE INDICATOR AND SAVE IT.
001710' 016767  176100  176362                                                  MOV     DVID1,DIND
001716'                         LET DMOD :B= #0                  ;CLEAR MODULE DROP FLAG.
001716' 105067  176566                                                          CLRB    DMOD
001722'                         CALL IDP                         ;CALL IDP-INITIALIZE DEVICE POINTERS.
001722' 004767  001046                                                          JSR     PC,IDP
001726'                         WHILEB EOD EQ #0 DO              ;WHILE THERE ARE MORE DEVICES:
001726'                                                                 50000$:
001726' 105767  176554                                                          TSTB    EOD
001732' 001053                                                                  BNE     50001$
001734'
001734' 016701  176334                   LET R1 := DCNT          ;SAVE DEVICE COUNTER IN REGISTER,
                                                                                MOV     DCNT,R1
001740' 016702  176332                   LET R2 := DINX          ;SAVE DEVICE INDEX IN REGISTER,
                                                                                MOV     DINX,R2
001744' 010203                           LET R3 := R2 SHIFT 1    ;SAVE DEVICE ADDRESS INCREMENT IN REGISTER,
                                                                                MOV     R2,R3
001746' 006303                                                                  ASL     R3
001750'                                  LET TTSDB(R2) := ADDR + R3 ;GENERATE AND STORE TSDB ADR IN TSDB TABLE,
001750' 016762  176032  000322'                                                 MOV     ADDR,TTSDB(R2)
001756' 060362  000322'                                                         ADD     R3,TTSDB(R2)
001762'                                  LET TTSSR(R2) := TTSDB(R2) + #2 ;GENERATE AND STORE TSSR ADR IN TSSR TABLE,
001762' 016762  000322' 000332'                                                 MOV     TTSDB(R2),TTSSR(
001770' 062762  000002  000332'                                                 ADD     #2,TTSSR(R2)
001776'                                  IF R2 EQ #0 THEN        ;IF THIS IS THE FIRST DEVICE THEN:
001776' 005702                                                                  TST     R2
002000' 001004                                                                  BNE     50002$
002002'                                    LET TVECT := VECTOR     ;STORE VECTOR DEFAULT IN VECTOR TABLE.
002002' 016767  176002  176332                                                  MOV     VECTOR,TVECT
002010'                                  ELSE                    ;ELSE - FOR ALL OTHER DEVICES:
002010' 000403                                                                  BR      50003$
002012'                                                                 50002$:
002012'                                    LET TVECT(R2) := SR1(R2) ;STORE SR2-SR4 CONTENTS IN VECTOR TABLE.
002012' 016262  000016' 000342'                                                 MOV     SR1(R2),TVECT(R2
002020'                                  ENDIF
002020'                                                                 50003$:
002020'                                  LET TREC(R2) := #0      ;CLEAR THIS DEVICE'S RECORD COUNT.
002020' 005062  000352'                                                         CLR     TREC(R2)
002024'                                  LET UIFLO(R1) :B= #0    ;CLEAR UNEXPECTED INTERRUPT FLAG.
002024' 105061  000512'                                                         CLRB    UIFLO(R1)
002030'                                  CALL DVSET              ;CALL DVSET-SET UP DEVICE VARIABLES.
002030' 004767  000672                                                          JSR     PC,DVSET
002034'                                  LET R4 := VECT          ;SAVE VECTOR ADDRESS IN REGISTER,
002034' 016704  176246                                                          MOV     VECT,R4
002040'                                  LET (R4)+ := UIADR(R2)  ;POINT VECTOR TO UNEXPECTED INTERRUPT ROUTINE,
002040' 016224  000536'                                                         MOV     UIADR(R2),(R4)+
002044'                                  LET (R4) :B= BR1        ;STORE INTERRUPT PRIORITY IN VECTOR +2,
002044' 116714  175742                                                          MOVB    BR1,(R4)
002050'                                  LET @TSSR := #0         ;ISSUE SUBSYSTEM INITIALIZATION TO TS04,
002050' 005077  176230                                                          CLR     @TSSR
```

```
                                        CALL UPDP                       ;CALL UPDP-UPDATE DEVICE POINTER TO NEXT DEV.
                                                                                        JSR     PC,UPDP
                                        ENDDO                           ;END OF DEVICE INITIALIZATION LOOP.
                                                                                        BR      50000$
                                                                                50001$:
                                        BR      CONTUE                  ;BYPASS DT03 BUS SWITCH CHECK WHEN STARTING.
                                ;       ---------------------------------
                                        RESTRT: IF PASCNT EQ #0 THEN    ;IF THE PASS COUNT IS CLEARED THEN:
                                                                                        TST     PASCNT
                                                                                        BNE     50004$
                                          BR    START                   ;GO TO START, PROCESSORS HAVE BEEN SWITCHED.
                                        ENDIF                           ;END OF DT03 BUS SWITCH CHECK.
                                                                                50004$:
                                ;       ---------------------------------

                                        .NLIST  MC
                                        .LIST   ME

                                        CONTUE:
                                        GETPAS,BEGIN, RBUFVA            ;GET PHYSICAL ADDRESS FROM 16-BIT RBUFVA
                                        GETPAS,BEGIN, CBUFVA            ;GET PHYSICAL ADDRESS FROM 16-BIT CBUFVA
                                        .LIST   MC
                                        .NLIST  ME
                                        IF #BIT0!BIT1 SETIN CBUFPA THEN ;IF CMDPKT ADR IS NOT ON A MODULO 4 BOUNDRY THEN
                                                                                        BIT     #BIT0!BIT1,CBUFP
                                                                                        BEQ     50005$
                                          LET CBUFVA := CBUFVA + #2      ;INCREASE CMDPKT VIRTUAL ADR BY 2.
                                                                                        ADD     #2,CBUFVA
                                          .NLIST        MC
                                          .LIST         ME
                                        GETPAS,BEGIN, CBUFVA            ;GET PHYSICAL ADDRESS FROM 16-BIT CBUFVA
                                          .LIST         MC
                                          .NLIST        ME
                                        ENDIF                           ;END OF CMD PACKET ADDRESS CHECK.
                                                                                50005$:
                                        LET CMDADR := CBUFEA SHIFT -4   ;SHIFT EXTENDED ADR TO BITS 0+1,
                                                                                        MOV     CBUFEA,CMDADR
                                                                                        ASR     CMDADR
                                                                                        ASR     CMDADR
                                                                                        ASR     CMDADR
                                                                                        ASR     CMDADR
                                        LET CMDADR := CMDADR + CBUFPA   ;ADD PHYSICAL AND EXTENDED ADDRESSES
                                                                                        ADD     CBUFPA,CMDADR
                                                                        ;AND STORE AS CMD PACKET ADDRESS.
                                        LET CMDPK1 := CBUFVA            ;SAVE ADDRESS OF COMMAND PACKET 1ST WORD.
                                                                                        MOV     CBUFVA,CMDPK1
                                        LET CMDPK2 := CBUFVA + #2       ;SAVE ADDRESS OF COMMAND PACKET 2ND WORD.
                                                                                        MOV     CBUFVA,CMDPK2
                                                                                        ADD     #2,CMDPK2
                                        LET CMDPK3 := CBUFVA + #4       ;SAVE ADDRESS OF COMMAND PACKET 3RD WORD.
                                                                                        MOV     CBUFVA,CMDPK3
                                                                                        ADD     #4,CMDPK3
                                        LET CMDPK4 := CBUFVA + #6       ;SAVE ADDRESS OF COMMAND PACKET 4TH WORD.
                                                                                        MOV     CBUFVA,CMDPK4
                                                                                        ADD     #6,CMDPK4
                                        .NLIST  MC
```



```
                                        .LIST   ME
                                        GETPAS,BEGIN, MBUFVA            ;GET PHYSICAL ADDRESS FROM 16-BIT MBUFVA
                                        GETPAS,BEGIN, SBUFVA            ;GET PHYSICAL ADDRESS FROM 16-BIT SBUFVA
                                        .LIST   MC
                                        .NLIST  ME

                                                                        ;PHYSICAL ADDRESS FROM MONITOR.
                                        LET MSGBLK := MBUFPA            ;MOVE MESSAGE PACKET ADDRESS TO
                                                                                        MOV     MBUFPA,MSGBLK
                                                                        ;CHARACTERISTIC BLOCK 1ST WORD.
                                        LET MSGBLK+2 := MBUFEA SHIFT -4 ;SHIFT EXTENDED ADDRESS TO BITS 0+1.
                                                                                        MOV     MBUFEA,MSGBLK+2
                                                                                        ASR     MSGBLK+2
                                                                                        ASR     MSGBLK+2
                                                                                        ASR     MSGBLK+2
                                                                                        ASR     MSGBLK+2
                                                                        ;AND STORE IN CHARACTERISTIC BLOCK 2ND WD.
                                        CALL IDP                        ;CALL IDP-INITIALIZE DEVICE POINTERS.
                                                                                        JSR     PC,IDP
                                        WHILEB EOD EQ #0 DO             ;WHILE THERE ARE MORE DEVICES:
                                                                                50006$:
                                                                                        TSTB    EOD
                                                                                        BNE     50007$
                                          CALL DVSET                      ;CALL DVSET-SET UP DEVICE VARIABLES.
                                                                                        JSR     PC,DVSET
                                          JSR R5,CVSET                    ;CALL CVSET-SET UP COMMAND VARIABLES.
                                          .WORD SCHAR                     ;ADR OF SET CHAR CMD VARIABLE TABLE.
                                          CALL ICCS                       ;CALL ICCS-ISSUE SET CHARACTERISTICS
                                                                                        JSR     PC,ICCS
                                                                          ;COMMAND AND CHECK STATUS.
                                          IFB DROP EQ #0 THEN             ;IF CURRENT DEVICE NOT DROPPED THEN:
                                                                                        TSTB    DROP
                                                                                        BNE     50010$
                                            JSR R5,CVSET                    ;CALL CVSET-SET UP CMD VARIABLES.
                                            .WORD ERASE                     ;ADR OF ERASE CMD VARIABLE TABLE.
                                            CALL ICCS                       ;CALL ICCS-ISSUE ERASE COMMAND
                                                                                        JSR     PC,ICCS
                                                                            ;AND CHECK STATUS.
                                          ENDIF                           ;END OF DEVICE DROP CHECK.
                                                                                50010$:
                                          CALL UPDP                       ;CALL UPDP-UPDATE DEVICE POINTERS.
                                                                                        JSR     PC,UPDP
                                        ENDDO                           ;END OF SET CHARACTERISTIC LOOP.
                                                                                        BR      50006$
                                                                                50007$:
                                ;       ---------------------------------
```

```
                                        BEGIN MAIN                      ;BEGINNING OF MAIN MODULE.
                                                                                $MAIN:
                                        WHILE ICOUNT LT ICONT DO        ;WHILE CYCLE COUNT IS LESS THEN THE
                                                                                50001$:
002370- 026767  175444  175440                                                  CMP     ICOUNT,ICONT
002376- 002105                                                                  BGE     50002$
                                                                        ;NUMBER OF CYCLES PER PASS:
                                          IFB DMOD NE #0 THEN             ;IF DROP MODULE FLAG IS SET THEN:
002400- 105767  176104                                                          TSTB    DMOD
002404- 001402                                                                  BEQ     50003$
                                            INLINE <JMP FINIT>                          ;LEAVE MODULE MAIN FLOW-DROP MODULE.
002406- 000167  000204                                                          JMP     FINIT
                                          ENDIF                           ;END MODULE DROP CHECK.
002412-                                                                         50003$:
                                          .NLIST MC
                                          .LIST ME
002412- 104414  000000-                 GWBUF$,BEGIN            ;GET WRITE BUFFER INFORMATION
                                          .LIST MC
                                          .NLIST ME
                                          LET R2 := DINX                  ;SAVE DEVICE INDEX IN REGISTER,
002416-                                                                         MOV     DINX,R2
002416- 016702  175654
                                          LET TREC(R2) := TREC(R2) + #1   ;INCREMENT DEVICE RECORD COUNT,
002422-                                                                         INC     TREC(R2)
002422- 005262  000352-
                                          CALL DVSET                      ;CALL DVSET-SET UP DEVICE VARIABLES,
002426-                                                                         JSR     PC,DVSET
002426- 004767  000274
                                          JSR R5,CVSET                    ;CALL CVSET-SET UP COMMAND VARIABLES,
002432- 004567  000170
                                          .WORD WRITE                     ;ADDRESS OF WRITE CMD VARIABLE TABLE,
002436- 000416-
                                          CALL ICCS                       ;CALL ICCS-ISSUE WRITE CMD + CHECK STATUS,
002440-                                                                         JSR     PC,ICCS
002440- 004767  000640
                                          IFB DROP EQ #0 THEN             ;IF CURRENT DEVICE HAS NOT BEEN DROPPED THEN:
002444-                                                                         TSTB    DROP
002444- 105767  176037                                                          BNE     50004$
002450- 001053
                                            JSR R5,CVSET                    ;CALL CVSET-SET UP COMMAND VARIABLES,
002452- 004567  000150
                                            .WORD RREV                      ;ADDRESS OF READ REV CMD VARIABLE TABLE,
002456- 000434-
                                            CALL ICCS                       ;CALL ICCS-ISSUE READ REV + CHECK STATUS
002460-                                                                         JSR     PC,ICCS
002460- 004767  000620
                                            IFB DROP EQ #0 THEN             ;IF THIS DEVICE HAS NOT BEEN DROPPED THEN:
002464-                                                                         TSTB    DROP
002464- 105767  176017                                                          BNE     50005$
002470- 001043
                                              LET ERRTYP := #1                ;LET ERROR TYPE EQUAL DATA COMPARE,
002472-                                                                         MOV     #1,ERRTYP
002472- 012767  000001  175406
                                            .NLIST MC
                                            .LIST ME
002500- 104412  000000- 000126-         CDATA$,BEGIN,RBUFPA     ; REQUEST FOR MONITOR TO CHECK DATA
002506- 002510-                         .+2                     ; IF ERROR, CONTINUE
                                            .LIST MC
                                            .NLIST ME
002510- 004567  000112                      JSR R5,CVSET                    ;CALL CVSET-SET UP COMMAND VARIABLES,
002514- 000452-                             .WORD RFWD                      ;ADDRESS OF READ FWD CMD VARIABLE TABLE,
002516-                                     CALL ICCS                       ;CALL ICCS-ISSUE READ FWD + STATUS CHECK,
002516- 004767  000562                                                              JSR     PC,ICCS
002522-                                     IFB DROP EQ #0 THEN             ;IF DEVICE HAS NOT BEEN DROPPED THEN:
002522- 105767  175761                                                              TSTB    DROP
002526- 001024                                                                      BNE     50006$
002530-                                       LET ERRTYP := #1                ;LET ERROR TYPE EQUAL DATA COMPARE,
002530- 012767  000001  175350                                                      MOV     #1,ERRTYP
```



```
                                            .NLIST MC
                                            .LIST ME
002536- 104412  000000- 000126-         CDATA$,BEGIN,RBUFPA     ; REQUEST FOR MONITOR TO CHECK DATA
002544- 002546-                         .+2                     ; IF ERROR, CONTINUE
                                            .LIST MC
                                            .NLIST ME
002546-                                     IF #XSO.EOT SETIN XSTAT0 THEN   ;IF AT END OF TAPE THEN:
002546- 032767  000001  176076                                                      BIT     #XSO.EOT,XSTAT0
002554- 001411                                                                     BEQ     50007$
002556- 004567  000044                        JSR R5,CVSET                    ;CALL CVSET-SETUP CMD VARIABLES,
002562- 000470-                               .WORD REWIND                    ;ADDRESS OF REWIND CMD VARIABLE TABLE
002564-                                       CALL ICCS                       ;CALL ICCS-ISSUE REWIND, CHECK STATUS,
002564- 004767  000514                                                              JSR     PC,ICCS
002570-                                       LET R2 := DINX                  ;SAVE DEVICE INDEX IN REGISTER,
002570- 016702  175502                                                              MOV     DINX,R2
002574-                                       LET TREC(R2) := #0              ;CLEAR DEVICE RECORD COUNT,
002574- 005062  000352-                                                             CLR     TREC(R2)
002600-                                     ENDIF                           ;END OF END OF TAPE CHECK,
002600-                                                                             50007$:
002600-                                   ENDIF                           ;END DEVICE DROP CHECK,
002600-                                                                         50006$:
002600-                                 ENDIF                           ;END DEVICE DROP CHECK,
002600-                                                                         50005$:
002600-                               ENDIF                           ;END DEVICE DROP CHECK
002600-                                                                         50004$:
002600-                               CALL UPDP                       ;CALL UPDP-UPDATE DEVICE POINTERS,
002600- 004767  000220                                                          JSR     PC,UPDP
                                      .NLIST MC
                                      .LIST ME
002604- 104413  000000-               ENDIT$,BEGIN            ;SIGNAL END OF ITERATION.
                                                              ;MONITOR SHALL TEST END OF PASS
                                      .LIST MC
                                      .NLIST ME
002610-                               ENDDO                           ;END MODULE CYCLE LOOP.
002610- 000667                                                                  BR      50001$
002612-                                                                         50002$:
002612-                               END MAIN                        ;END MAIN MODULE.
                                ;     ----------------------------------
                                      BEGIN FINI                      ;BEGINNING OF FINI MODULE.
002616-                               INLINE <FINIT:>
002616-                                                                         FINIT:
                                      .NLIST  MC
                                      .LIST   ME
002616- 104410  000000-               ENDS,BEGIN              ;
                                      .LIST   MC
                                      .NLIST  ME
002622-                               END FINI                        ;END OF FINI MODULE.
                                ;     ----------------------------------
```

```
;*****************************************************************************
;*                                                                           *
;*SUBROUTINE NAME:       CVSET.                                             *
;*                                                                           *
;*FUNCTION:              SETS UP COMMAND PACKET AND OTHER CMD VARIABLES.     *
;*                                                                           *
;*CALLING SEQUENCE:      JSR     R5,CVSET                                   *
;*                       .WORD   TABLE                                      *
;*                                                                           *
;*PARAMETERS PASSED:     INPUT - THE LOCATION FOLLOWING THE CALL CONTAINS    *
;*                       THE ADDRESS OF A TABLE CONTAINING ALL COMMAND       *
;*                       VARIABLES. AFTER THE LINK REGISTER (R5) IS USED     *
;*                       TO LOCATE THE TABLE, THE REGISTER MUST BE IN-       *
;*                       CREMENTED BY 2 TO ENABLE THE RETURN.                *
;*                                                                           *
;*                       COMMAND VARIABLE TABLE CONTENTS:                    *
;*                       WORD 1 - TS04 COMMAND                               *
;*                            2 - TS04 RETRY COMMAND                         *
;*                            3 - RETRY LIMIT                                *
;*                            4 - ADDRESS OF COMMAND ASCII                   *
;*                            5 - ADDRESS OF BUFFER PHYSICAL ADR             *
;*                            6 - ADDRESS OF BUFFER EXTENDED ADR             *
;*                            7 - ADDRESS OF BUFFER SIZE                     *
;*                                                                           *
;*REGISTERS USED:        R4, R5.                                            *
;*                                                                           *
;*SUBORDINATE ROUTINES: NONE                                                *
;*                                                                           *
;*****************************************************************************

002626' CVSET:  PUSH R4                         ;SAVE TABLE INDEX REGISTER ON THE STACK.
                                                                    MOV     R4,-(SP)
002626' 010446
002630'         LET R4 := (R5)+                 ;LET INDEX REGISTER EQUAL THE ADDRESS
                                                                    MOV     (R5)+,R4
002630' 012504                                  ;OF THE CMD VARIABLE TABLE AND INCREMENT
                                                ;LINK REGISTER FOR RETURN.
002632'         LET @CMDPK1 := (R4)             ;MOVE CMD WORD TO CMD PACKET 1ST WORD.
                                                                    MOV     (R4),@CMDPK1
002632' 011477  175712
002636'         LET CMD := (R4)+                ;SAVE THE CMD WORD FOR REFERENCE.
                                                                    MOV     (R4)+,CMD
002636' 012467  175450
002642'         LET RCMD := (R4)+               ;GET RETRY CMD FROM TABLE AND SAVE IT.
                                                                    MOV     (R4)+,RCMD
002642' 012467  175446
002646'         LET RTRYL := (R4)+              ;GET RETRY LIMIT FROM TABLE AND SAVE IT
                                                                    MOV     (R4)+,RTRYL
002646' 012467  175444
002652'         LET CMDA := (R4)+               ;GET CMD ASCII ADR FROM TABLE AND SAVE IT.
                                                                    MOV     (R4)+,CMDA
002652' 012467  002330
002656'         LET @CMDPK2 := @(R4)+           ;MOVE BUFFER ADR TO CMD PACKET 2ND WORD.
                                                                    MOV     @(R4)+,@CMDPK2
002656' 013477  175670
002662'         LET @CMDPK3 := @(R4)+ SHIFT -4  ;MOVE EXTENDED ADR TO CMD PACKET 3RD WORD
                                                                    MOV     @(R4)+,@CMDPK3
002662' 013477  175666                                              ASR     @CMDPK3
002666' 006277  175662                                              ASR     @CMDPK3
002672' 006277  175656                                              ASR     @CMDPK3
002676' 006277  175652                                              ASR     @CMDPK3
002702' 006277  175646
                                                ;AND SHIFT TO BITS 0+1.
002706'         LET @CMDPK4 := @(R4)+ SHIFT 1   ;MOVE BUFFER SIZE TO CMD PACKET 4TH WORD
```



```
002706' 013477  175644                                              MOV     @(R4)+,@CMDPK4
002712' 006377  175640                                              ASL     @CMDPK4
                                                ;AND DOUBLE FOR BYTE COUNT.
002716'         LET RTRYC := #0                 ;CLEAR THE RETRY COUNT.
002716' 005067  175376                                              CLR     RTRYC
002722'         POP R4                          ;RESTORE THE TABLE INDEX REGISTER.
                                                                    MOV     (SP)+,R4
002722' 012604
002724' 000205  RTS     R5                      ;RETURN
;       ---------------------------------
```

```
                        ;*********************************************************************
                        ;*                                                                   *
                        ;*SUBROUTINE NAME:      DVSET                                        *
                        ;*                                                                   *
                        ;*FUNCTION:             SETS UP ALL DEVICE VARIABLES.                *
                        ;*                                                                   *
                        ;*CALLING SEQUENCE:     CALL DVSET.                                  *
                        ;*                                                                   *
                        ;*PARAMETERS PASSED:    NONE                                         *
                        ;*                                                                   *
                        ;*REGISTERS USED:       R2.                                          *
                        ;*                                                                   *
                        ;*SUBORDINATE ROUTINES: NONE                                         *
                        ;*                                                                   *
                        ;*********************************************************************

002726'                 DVSET:  PUSH R2                         ;SAVE TABLE INDEX REGISTER ON STACK.
002726' 010246                          MOV     R2,-(SP)
002730'                         LET R2 := DINX                  ;LET INDEX REGISTER EQUAL DEVICE INDEX.
002730' 016702  175342                  MOV     DINX,R2
002734'                         LET CSRA := TTSDB(R2)           ;GET TSDB ADR FROM TSDB TABLE AND SAVE IT.
002734' 016267  000322' 175136          MOV     TTSDB(R2),CSRA
002742'                         LET TSSR := TTSSR(R2)           ;GET TSSR ADR FROM TSSR TABLE AND SAVE IT.
002742' 016267  000332' 175334          MOV     TTSSR(R2),TSSR
002750'                         LET VECT := TVECT(R2)           ;GET VECTOR FROM VECTOR TABLE AND SAVE IT.
002750' 016267  000342' 175330          MOV     TVECT(R2),VECT
002756'                         LET RECORD := TREC(R2)          ;GET RECORD COUNT FROM TABLE AND SAVE IT.
002756' 016267  000352' 175324          MOV     TREC(R2),RECORD
002764'                         LET EOD :B= #0                  ;CLEAR END OF DEVICE FLAG.
002764' 105067  175516                  CLRB    EOD
002770'                         POP R2                          ;RESTORE THE TABLE INDEX REGISTER.
002770' 012602                          MOV     (SP)+,R2
002772' 000207                  RTS PC                          ;RETURN.
                        ;       ---------------------------------
```



```
                        ;*********************************************************************
                        ;*                                                                   *
                        ;*SUBROUTINE NAME:      IDP                                          *
                        ;*                                                                   *
                        ;*FUNCTION:             INITIALIZES DEVICE POINTERS.                 *
                        ;*                                                                   *
                        ;*                      DEFINITION OF DEVICE POINTERS:               *
                        ;*                       1. DEVICE COUNTER-COUNTS 0-3.               *
                        ;*                       2. DEVICE INDEX-COUNTS 0-6 BY 2'S           *
                        ;*                       3. VARIABLE DEVICE INDICATOR-CONTAINS       *
                        ;*                          ONLY THE BIT OF THE CURRENT DEVICE       *
                        ;*                          FROM THE DEVICE INDICATOR IN THE HEADER. *
                        ;*                                                                   *
                        ;*                      +-----------------------------+              *
                        ;*                      | DEVICE        | 0 | 1 | 2 | 3 |              *
                        ;*                      |---------------+---+---+---+---|              *
                        ;*                      | COUNTER       | 0 | 1 | 2 | 3 |              *
                        ;*                      |---------------+---+---+---+---|              *
                        ;*                      | INDEX         | 0 | 2 | 4 | 6 |              *
                        ;*                      |---------------+---+---+---+---|              *
                        ;*                      | VARIABLE IND. | 1 | 2 | 4 | 8 |              *
                        ;*                      +-----------------------------+              *
                        ;*                                                                   *
                        ;*CALLING SEQUENCE:     CALL IDP                                     *
                        ;*                                                                   *
                        ;*PARAMETERS PASSED:    NONE                                         *
                        ;*                                                                   *
                        ;*REGISTERS USED:       NONE                                         *
                        ;*                                                                   *
                        ;*SUBORDINATE ROUTINES: NONE                                         *
                        ;*                                                                   *
                        ;*********************************************************************

002774'                 IDP:    LET DCNT := #0                  ;CLEAR DEVICE COUNTER.
002774' 005067  175274                  CLR     DCNT
003000'                         LET DINX := #0                  ;CLEAR DEVICE INDEX.
003000' 005067  175272                  CLR     DINX
003004'                         LET VDIND := #1                 ;LET VARIABLE DEVICE INDICATOR EQUAL 1.
003004' 012767  000001  175270          MOV     #1,VDIND
003012'                         LET EOD :B= #0                  ;CLEAR END OF DEVICES FLAG.
003012' 105067  175470                  CLRB    EOD
003016'                         LET DROP :B= #0                 ;CLEAR DEVICE DROPPED FLAG.
003016' 105067  175465                  CLRB    DROP
003022' 000207                  RTS PC                          ;RETURN.
                        ;       ---------------------------------
```

```
                        ;*********************************************************************
                        ;*                                                                   *
                        ;*SUBROUTINE NAME:      UPDP                                         *
                        ;*                                                                   *
                        ;*FUNCTION:             UPDATE DEVICE POINTERS TO NEXT DEVICE.       *
                        ;*                                                                   *
                        ;*CALLING SEQUENCE:     CALL UPDP                                    *
                        ;*                                                                   *
                        ;*PARAMETERS PASSED:    NONE                                         *
                        ;*                                                                   *
                        ;*REGISTERS USED:       R2                                           *
                        ;*                                                                   *
                        ;*SUBORDINATE ROUTINES: DROPD                                        *
                        ;*                                                                   *
                        ;*********************************************************************

003024'                 UPDP:   PUSH R2                         ;SAVE TABLE INDEX REGISTER ON STACK.
003024' 010246                                                                  MOV     R2,-(SP)
003026'                         LET R2 := DCNT                  ;LET IDNEX REGISTER EQUAL DEVICE COUNTER
003026' 016702  175242                                                          MOV     DCNT,R2
003032'                         IFB UIFLO(R2) NE #0 THEN        ;IF AN UNEXPECTED INTERRUPT HAS OCCURRED THEN:
003032' 105762  000512'                                                         TSTB    UIFLO(R2)
003036' 001424                                                                  BEQ     50011$
003040'                           LET ERRTYP := #14               ;LET ERROR TYPE EQUAL 14,
003040' 012767  000014  175040                                                  MOV     #14,ERRTYP
003046'                           LET ASTAT := @TSSR              ;SAVE CONTENTS OF TSSR FOR PRINTOUT.
003046' 017767  175232  175030                                                  MOV     @TSSR,ASTAT
003054'                           LET ACSR := @CSRA               ;SAVE CONTENTS OF TSDB(CSR) FOR PRINTOUT.
003054' 017767  175020  175020                                                  MOV     @CSRA,ACSR
                                  .NLIST MC
                                  .LIST ME
003062' 104403  000000' 005142'         MSGMS$BEGIN_UNEX        ;ASCII MESSAGE CALL WITH COMMON HEADER
003070' 104403  000000' 005206'         MSGMS$BEGIN_CMDA        ;ASCII MESSAGE CALL WITH COMMON HEADER
                                        ;*************************************************************
003076' 104405  000000' 000570'         HRDER$_BEGIN_TABLE      ;UNEXPECTED INTERRUPT
                                        ;*************************************************************
                                  .LIST MC
                                  .NLIST ME
003104'                           CALL DROPD                    ;CALL DROPD-DROP THE CURRENT DEVICE.
003104' 004767  000126                                                          JSR     PC,DROPD
003110'                         ENDIF                           ;END OF UNEXPECTED INTERRUPT CHECK.
003110'                                                                 50011$:
003110'                         LET DROP :B= #0                 ;CLEAR DEVICE DROPPED FLAG.
003110' 105067  175373                                                          CLRB    DROP
003114'                         REPEAT                          ;REPEAT UNTIL AN ACTIVE DEVICE IS FOUND:
003114'                                                                 50012$:
003114'                           LET DCNT := DCNT + #1           ;INCREMENT DEVICE COUNTER,
003114' 005267  175154                                                          INC     DCNT
003120'                           IF DCNT GT #3 THEN              ;IF DEVICE COUNT GREATER THAN 3 THEN:
003120' 026727  175150  000003                                                  CMP     DCNT,#3
003126' 003423                                                                  BLE     50013$
003130'                             IFB EOD NE #0 THEN            ;IF END OF DEVICES FLAG IS SET THEN:
003130' 105767  175352                                                          TSTB    EOD
003134' 001405                                                                  BEQ     50014$
003136'                               LET DMOD :B= #1             ;SET DROP MODULE FLAG,
003136' 112767  000001  175344                                                  MOVB    #1,DMOD
```



```
003144' 000432                            BR UPDRTN             ;BRANCH TO RETURN-ALL DEVICES DROPPED.
003146'                                 ELSE                    ;ELSE-IF END OF DEVICES NOT SET:
003146' 000412                                                                  BR      50015$
003150'                                                                 50013$:
003150'                                   LET EOD :B= #1          ;SET END OF DEVICES FLAG,
003150' 112767  000001  175330                                                  MOVB    #1,EOD
003156'                                   LET DCNT := #0          ;CLEAR DEVICE COUNTER,
003156' 005067  175112                                                          CLR     DCNT
003162'                                   LET DINX := #0          ;CLEAR DEVICE INDEX,
003162' 005067  175110                                                          CLR     DINX
003166'                                   LET VDIND := #1         ;SET VARIABLE DEV INDICATOR TO 1.
003166' 012767  000001  175106                                                  MOV     #1,VDIND
003174'                                 ENDIF                   ;END OF DEVICE COUNTER WRAP AROUND CHECK.
003174'                                                                 50015$:
003174'                               ELSE                      ;ELSE-IF DEVICE COUNT IS 3 OR LESS:
003174' 000412                                                                  BR      50016$
003176'                                                                 50013$:
003176'                                   LET DINX := DCNT SHIFT 1  ;LET DEV INDEX EQUAL DEV COUNT X 2,
003176' 016767  175072  175072                                                  MOV     DCNT,DINX
003204' 006367  175066                                                          ASL     DINX
003210'                                   LET R2 := DINX            ;LET INDEX REGISTER EQUAL DEVICE INDEX,
003210' 016702  175062                                                          MOV     DINX,R2
003214'                                   LET VDIND := BCON(R2)     ;LET THE VARIABLE DEVICE INDICATOR EQUAL
003214' 016267  000560'  175060                                                 MOV     BCON(R2),VDIND
                                                                        ;THE APPROPRIATE BIT.
                                        ENDIF                   ;END OF DEVICE POINTER SET UP.
                                                                        50016$:
003222'                         UNTIL VDIND SETIN DIND          ;END OF DEVICE SEARCH REPEAT LOOP.
003222' 036767  175054  175050                                                  BIT     VDIND,DIND
003230' 001731                                                                  BEQ     50012$
003232'                 UPDRTN: POP R2                          ;RESTORE TABLE INDEX REGISTER.
003232' 012602                                                                  MOV     (SP)+,R2
003234' 000207                  RTS PC                          ;RETURN.
                        ;       ----------
```

```
                        ;*********************************************************************
                        ;*                                                                   *
                        ;*SUBROUTINE NAME:      DROPD                                        *
                        ;*                                                                   *
                        ;*FUNCTION:             DROPS THE CURRENT DEVICE FROM THE TEST CYCLE. *
                        ;*                                                                   *
                        ;*CALLING SEQUENCE:     CALL DROPD                                   *
                        ;*                                                                   *
                        ;*PARAMETERS PASSED:    NONE                                         *
                        ;*                                                                   *
                        ;*REGISTERS USED:       R2                                           *
                        ;*                                                                   *
                        ;*SUBORDINATE ROUTINES: NONE.                                        *
                        ;*                                                                   *
                        ;*********************************************************************

003236' 010246          DROPD:  PUSH R2                         ;SAVE TABLE INDEX REGISTER ON THE STACK.
                                                                        MOV     R2,-(SP)
003240' 016702  175032          LET R2 := DINX                  ;LET INDEX REGISTER EQUAL DEVICE INDEX.
                                                                        MOV     DINX,R2
003244' 016277  000536' 175034  LET @VECT := UIADR(R2)           ;POINT VECTOR TO UNEXPECTED INTERRUPT ROUTINE.
                                                                        MOV     UIADR(R2),@VECT
003252' 005077  175026          LET @TSSR := #0                 ;ISSUE SUBSYSTEM INITIALIZATION TO TS04.
                                                                        CLR     @TSSR
003256' 046767  175020  175014  LET DIND := DIND CLR.BY VDIND   ;CLEAR THIS DEVICE'S BIT IN DEVICE INDICATOR.
                                                                        BIC     VDIND,DIND
003264' 112767  000001  175215  LET DROP :B= #1                 ;SET DEVICE DROPPED FLAG.
                                                                        MOVB    #1,DROP
                                .NLIST MC
                                .LIST   ME
003272' 104403  000000' 005202' MSGNS,BEGIN,DROE        ;ASCII MESSAGE CALL WITH COMMON HEADER
                                .LIST   MC
                                .NLIST  ME
003300'                         POP R2                          ;RESTORE TABLE INDEX REGISTER.
003300' 012602                                                          MOV     (SP)+,R2
003302' 000207                  RTS PC                          ;RETURN.
                        ;       ----------
```



```
                        ;*********************************************************************
                        ;*                                                                   *
                        ;*SUBROUTINE NAME:      ICCS                                         *
                        ;*                                                                   *
                        ;*FUNCTIONS:            - ISSUES COMMAND PACKET TO TS04.             *
                        ;*                      - WAITS FOR INTERRUPT.                       *
                        ;*                      - CHECKS STATUS.                             *
                        ;*                      - EXECUTES ERROR RECOVERY.                   *
                        ;*                                                                   *
                        ;*CALLING SEQUENCE:     CALL ICCS                                    *
                        ;*                                                                   *
                        ;*PARAMETERS PASSES:    NONE                                         *
                        ;*                                                                   *
                        ;*REGISTERS USED:       R2                                           *
                        ;*                                                                   *
                        ;*SUBORDINATE ROUTINES: DROPD, TCC0, TCC1, TCC2, TCC3, TCC4,         *
                        ;*                      TCC5, TCC6, TCC7.                            *
                        ;*                                                                   *
                        ;*ENTRY PREREQUISITES:  - ALL DEVICE VARIABLES SET UP.               *
                        ;*                      - ALL COMMAND VARIABLES SET UP.              *
                        ;*                                                                   *
                        ;*********************************************************************

003304' 010246          ICCS:   PUSH R2                         ;SAVE TABLE INDEX REGISTER ON STACK.
003304'                                                                 MOV     R2,-(SP)
003306' 105067  175177  ICCSR:  LET EREC :B= #0                  ;CLEAR ER FLAG.
                                                                        CLRB    EREC
003312' 012767  037777  175300  LET TIME := #37777               ;INITIATE TIMEOUT COUNTER.
                                                                        MOV     #37777,TIME
003320'                         REPEAT                          ;REPEAT UNTIL DEVICE IS READY:
                                                                                50017$:
                                  .NLIST MC
                                  .LIST   ME
003320' 104407  000000'         BREAKS,BEGIN            ;TEMPORARY RETURN TO MONITOR....
003324' 104407  000000'         BREAKS,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
                                  .LIST MC
                                  .NLIST ME
003330'                           IF TIME EQ #0 THEN              ;IF TIMEOUT HAS OCCURRED THEN:
003330' 005767  175264                                                  TST     TIME
003334' 001025                                                          BNE     50020$
003336'                             LET ERRTYP := #3                ;LET ERROR TYPE EQUAL 3,
003336' 012767  000003  174542                                          MOV     #3,ERRTYP
003344'                             LET ASTAT := @TSSR              ;SAVE CONTENTS OF TSSR FOR PRINTOUT.
003344' 017767  174734  174532                                          MOV     @TSSR,ASTAT
003352'                             LET ACSR := @CSRA               ;SAVE CONTENTS OF TSDB(CSR) FOR PRINTOUT.
003352' 017767  174522  174522                                          MOV     @CSRA,ACSR
                                    .NLIST MC
                                    .LIST ME
003360' 104403  000000' 005132' MSGNS,BEGIN,CONT        ;ASCII MESSAGE CALL WITH COMMON HEADER
003366' 104403  000000' 005206' MSGNS,BEGIN,CMDA        ;ASCII MESSAGE CALL WITH COMMON HEADER
                                ;*********************************************************************
003374' 104405  000000' 000570' HRDERS,BEGIN,TABLE      ;CONTROLLER NOT READY
                                ;*********************************************************************
                                  .LIST MC
                                  .NLIST ME
003402'                           CALL DROPD                    ;CALL DROPD-DROP CURRENT DEVICE,
```

```
1114  003402' 004767  177630                                                    JSR     PC,DROPD
1115  003406' 000476                          BR ICCRTN                       ;BRANCH TO ICCRTN-RETURN.
1116  003410'                                 ENDIF                           ;END OF READY TIMEOUT PROCESSING,
1117  003410'                                                                                 50020$:
1118  003410'                                 LET TIME := TIME - #1           ;UPDATE THE TIMEOUT COUNTER,
1119  003410' 005367  175204                                                    DEC     TIME
1120  003414'                                 UNTIL #TS.SSR SETIN @TSSR       ;END OF READY WAIT/REPEAT LOOP.
1121  003414' 032777  000200  174662                                            BIT     #TS.SSR,@TSSR
1122  003422' 001736                                                            BEQ     50017$
1123  003424'
1124  003424' 012777  003530' 174654          LET @VECT := #ICCS1              ;POINT VECTOR TO PIRQ CALL.
1125  003432'                                                                   MOV     #ICCS1,@VECT
1126  003432' 016777  175110  174440          LET @CSRA := CMDADR              ;ISSUE THE CMD PACKET TO THE TS04.
1127  003440' 000240                                                            MOV     CMDADR,@CSRA
1128  003442'                                 NOP                             ;GIVE TS04 ONE INSTUCTION TIME TO RESPOND.
1129  003442' 032777  000200  174634          IF #TS.SSR SETIN @TSSR THEN     ;IF READY DID NOT DROP THEN:
1130  003450' 001425                                                            BIT     #TS.SSR,@TSSR
1131  003452'                                                                   BEQ     50021$
1132  003452' 012767  000003  174426          LET ERRTYP := #3                 ;LET ERROR TYPE EQUAL 3.
1133  003460'                                                                   MOV     #3,ERRTYP
1134  003460' 017767  174620  174416          LET ASTAT := @TSSR               ;SAVE CONTENTS OF TSSR FOR PRINTOUT.
1135  003466'                                                                   MOV     @TSSR,ASTAT
1136  003466' 017767  174406  174406          LET ACSR := @CSRA                ;SAVE CONTENTS OF TSDB(CSR) FOR PRINTOUT.
1137                                                                            MOV     @CSRA,ACSR
1138                                          .NLIST MC
1139                                          .LIST ME
1140  003474' 104403  000000' 005136' MSGNS,BEGIN,READ       ;ASCII MESSAGE CALL WITH COMMON HEADER
1141  003502' 104403  000000' 005206' MSGNS,BEGIN,CMDA       ;ASCII MESSAGE CALL WITH COMMON HEADER
1142                                  ;*********************************************************
1143  003510' 104405  000000' 000570' ERRORS,BEGIN,TABLE     ;READY DID NOT DROP
1144                                  ;*********************************************************
1145                                  .LIST MC
1146                                  .NLIST ME
1147  003516'                         CALL DROPD                      ;CALL DROPD-DROP CURRENT DEVICE.
1148  003516' 004767  177514                                                    JSR     PC,DROPD
1149  003522' 000430                  BR ICCRTN                       ;BRANCH TO ICCRTN-RETURN.
1150  003524'                         ENDIF                           ;END OF READY DROP PROCESSING,
1151  003524'                                                                                 50021$:
1152                                  .NLIST MC
1153                                  .LIST ME
1154  003524' 104400  000000'         EXITS,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
1155  003530'                 ICCS1:  ;---------------------------------------------------------------
1156  003530' 000004  000000' 003536' PIRQS,BEGIN,ICCS2               ; QUEUE UP TO CONTINUE AT ICCS2 AND RTI
1157                                  ;---------------------------------------------------------------
1158                                  .LIST MC
1159                                  .NLIST ME
1160  003536'                 ICCS2:  LET R2 := DINX                   ;LET INDEX REGISTER EQUAL DEVICE INDEX.
1161  003536' 016702  174534                                                    MOV     DINX,R2
1162  003542'                         LET @VECT := UIADR(R2)           ;POINT VECTOR TO UNEXPECTED INTERRUPT ROUTINE.
1163  003542' 016277  000536' 174536                                            MOV     UIADR(R2),@VECT
1164  003550'                         IF #TS.SC SETIN @TSSR THEN       ;IF SPECIAL CONDITION IS SET IN TSSR THEN:
1165  003550' 032777  100000  174526                                            BIT     #TS.SC,@TSSR
1166  003556' 001406                                                            BEQ     50022$
1167  003560'                           LET R2 := @TSSR CLR.BY #177761  ;MASK TERMINATION CODE OUT OF TSSR.
1168  003560' 017702  174520                                                    MOV     @TSSR,R2
1169  003564' 042702  177761                                                    BIC     #177761,R2
```



```
1170  003570' 004772  000516'           JSR PC,@TCCRA(R2)               ;GO TO APPROPRIATE TCC HANDLING ROUTINE.
1171  003574'                         ENDIF                           ;END OF SPECIAL CONDITION STATUS CHECK.
1172  003574'                                                                                 50022$:
1173  003574'                         IFB EREC NE #0 THEN             ;IF ERROR RECOVERY REQUIRED FLAG IS SET THEN:
1174  003574' 105767  174711                                                    TSTB    EREC
1175  003600' 001401                                                            BEQ     50023$
1176  003602' 000641                   BR ICCSR                       ;GO DO ERROR RECOVERY.
1177  003604'                         ENDIF                           ;END OF ERROR RECOVERY REQUIREMENT CKECK.
1178  003604'                                                                                 50023$:
1179  003604'                 ICCRTN: POP R2                          ;RESTORE TABLE INDEX REGISTER.
1180  003604' 012602                                                            MOV     (SP)+,R2
1181  003606' 000207                  RTS PC                          ;RETURN.
1182                                  ;
1183
```

```
                        ;*********************************************************************
                        ;*                                                                   *
                        ;*SUBROUTINE NAME:      TCC0                                         *
                        ;*                                                                   *
                        ;*FUNCTION:             PROCESSES TERMINATION CLASS CODE 0           *
                        ;*                      (UNDEFINED SPECIAL CONDITION STATUS).        *
                        ;*                                                                   *
                        ;*CALLING SEQUENCE:     CALL TCC0                                    *
                        ;*                                                                   *
                        ;*PARAMETERS PASSES:    NONE                                         *
                        ;*                                                                   *
                        ;*REGISTERS USED:       NONE                                         *
                        ;*                                                                   *
                        ;*SUBORDINATE ROUTINES: DROPD                                        *
                        ;*                                                                   *
                        ;*ENTRY PREREQUISITES:  - COMMAND HAS BEEN ISSUED TO TS04.           *
                        ;*                      - INTERRUPT HAS BEEN RECEIVED.               *
                        ;*                                                                   *
                        ;*********************************************************************

003610'                 TCC0:   LET ERRTYP := #0                ;LET ERROR TYPE EQUAL 0,    CLR     ERRTYP
003610' 005067  174272          LET ASTAT := @TSSR              ;SAVE CONTENTS OF TSSR FOR PRINTOUT,
003614' 017767  174464  174262                                                              MOV     @TSSR,ASTAT
003622' 017767  174252  174252  LET ACSR := @CSRA                ;SAVE CONTENTS OF TSDB(CSR) FOR PRINTOUT.
003622'                                                                                     MOV     @CSRA,ACSR
                                .NLIST  ME
                                .LIST   ME
003630' 104403  000000' 005145' MSGMS,BEGIN,SPEC        ;ASCII MESSAGE CALL WITH COMMON HEADER
003636' 104403  000000' 005206' MSGMS,BEGIN,CMDA        ;ASCII MESSAGE CALL WITH COMMON HEADER
                                ;*****************************************************************
003644' 104406  000000' 000570' SOFERS,BEGIN,TABLE      ;SPECIAL CONDITION STATUS
                                ;*****************************************************************
                                .LIST   ME
                                .NLIST  ME
003652'                         CALL DROPD                      ;CALL DROPD-DROP THE DEVICE.
003652' 004767  177360                                                                      JSR     PC,DROPD
003656' 000207                  RTS PC                          ;RETURN.
                        ;       ---------------------------------
```



```
                        ;*********************************************************************
                        ;*                                                                   *
                        ;*SUBROUTINE NAME:      TCC1                                         *
                        ;*                                                                   *
                        ;*FUNCTION:             PROCESSES TCC1 (ATTENTION CONDITION).        *
                        ;*                                                                   *
                        ;*CALLING SEQUENCE:     CALL TCC1                                    *
                        ;*                                                                   *
                        ;*PARAMETERS PASSES:    NONE                                         *
                        ;*                                                                   *
                        ;*REGISTERS USED:       NONE                                         *
                        ;*                                                                   *
                        ;*SUBORDINATE ROUTINES: DROPD                                        *
                        ;*                                                                   *
                        ;*ENTRY PREREQUISITES:  - COMMAND HAS BEEN ISSUED TO TS04.           *
                        ;*                      - INTERRUPT HAS BEEN RECEIVED.               *
                        ;*                                                                   *
                        ;*********************************************************************

003660'                 TCC1:   LET ERRTYP := #6                ;SET ERROR TYPE EQUAL 6,    MOV     #6,ERRTYP
003660' 012767  000006  174220  LET ASTAT := @TSSR              ;SAVE CONTENTS OF TSSR FOR PRINTOUT,
003666' 017767  174412  174210                                                              MOV     @TSSR,ASTAT
003674' 017767  174200  174200  LET ACSR := @CSRA                ;SAVE CONTENTS OF TSDB(CSR) FOR PRINTOUT.
003674'                                                                                     MOV     @CSRA,ACSR
                                .NLIST  ME
                                .LIST   ME
003702' 104403  000000' 005152' MSGMS,BEGIN,DEVI        ;ASCII MESSAGE CALL WITH COMMON HEADER
003710' 104403  000000' 005206' MSGMS,BEGIN,CMDA        ;ASCII MESSAGE CALL WITH COMMON HEADER
                                ;*****************************************************************
003716' 104405  000000' 000570' HRDERS,BEGIN,TABLE      ;DEVICE OFF LINE
                                ;*****************************************************************
                                .LIST   ME
                                .NLIST  ME
003724'                         CALL DROPD                      ;CALL DROPD-DROP THE DEVICE.
003724' 004767  177306                                                                      JSR     PC,DROPD
003730' 000207                  RTS PC                          ;RETURN.
                        ;       ---------------------------------
```

```
                        ;*********************************************************************
                        ;*                                                                   *
                        ;*SUBROUTINE NAME:      TCC2                                         *
                        ;*                                                                   *
                        ;*FUNCTION:             PROCESSES TCC2 (TAPE STATUS ALERT).          *
                        ;*                                                                   *
                        ;*CALLING SEQUENCE:     CALL TCC2                                    *
                        ;*                                                                   *
                        ;*PARAMETERS PASSES:    NONE                                         *
                        ;*                                                                   *
                        ;*REGISTERS USED:       NONE                                         *
                        ;*                                                                   *
                        ;*SUBORDINATE ROUTINES: NONE                                         *
                        ;*                                                                   *
                        ;*ENTRY PREREQUISITES:  - COMMAND HAS BEEN ISSUED TO TS04.           *
                        ;*                      - INTERRUPT HAS BEEN RECEIVED.               *
                        ;*                                                                   *
                        ;*********************************************************************

003732'                 TCC2:   IF #XS0.EOT NOTSETIN XSTAT0 THEN  ;IF NOT AT END OF TAPE THEN:
003732' 032767  000001  174712                                                      BIT     #XS0.EOT,XSTAT0
003740' 001022                                                                      BNE     50024$
003742'                           LET ERRTYP := #51               ;LET ERROR TYPE EQUAL 51.
003742' 012767  000051  174136                                                      MOV     #51,ERRTYP
003750'                           LET ASTAT := @TSSR              ;SAVE CONTENTS OF TSSR FOR PRINTOUT.
003750' 017767  174330  174126                                                      MOV     @TSSR,ASTAT
003756'                           LET ACSR := @CSRA               ;SAVE CONTENTS OF TSDB(CSR) FOR PRINTOUT.
003756' 017767  174116  174116                                                      MOV     @CSRA,ACSR
                                  .NLIST MC
                                  .LIST ME
003764' 104403  000000' 005156'         MSGMS,BEGIN,TAPE        ;ASCII MESSAGE CALL WITH COMMON HEADER
003772' 104403  000000' 005206'         MSGMS,BEGIN,CMDA        ;ASCII MESSAGE CALL WITH COMMON HEADER
                                        ;*************************************************************
004000' 104406  000000' 000570'         SOFERS,BEGIN,TABLE      ;TAPE STATUS ALERT
                                        ;*************************************************************
                                  .LIST MC
                                  .NLIST ME
004006'                           ENDIF                                             ;END TAPE STATUS ALERT PROCESSING.
004006'                                                                     50024$:
004006' 000207                    RTS PC                          ;RETURN
                        ;         ---------------------------------
```



```
                        ;*********************************************************************
                        ;*                                                                   *
                        ;*SUBROUTINE NAME:      TCC3                                         *
                        ;*                                                                   *
                        ;*FUNCTION:             PROCESSES TCC3 (FUNCTION REJECT).            *
                        ;*                                                                   *
                        ;*CALLING SEQUENCE:     CALL TCC3                                    *
                        ;*                                                                   *
                        ;*PARAMETERS PASSES:    NONE                                         *
                        ;*                                                                   *
                        ;*REGISTERS USED:       NONE                                         *
                        ;*                                                                   *
                        ;*SUBORDINATE ROUTINES: DROPD                                        *
                        ;*                                                                   *
                        ;*ENTRY PREREQUISITES:  - COMMAND HAS BEEN ISSUED TO TS04.           *
                        ;*                      - INTERRUPT HAS BEEN RECEIVED                *
                        ;*                                                                   *
                        ;*********************************************************************

004010'                 TCC3:   LET ERRTYP := #52               ;LET ERROR TYPE EQUAL 52.
004010' 012767  000052  174070                                                      MOV     #52,ERRTYP
004016'                         LET ASTAT := @TSSR              ;SAVE CONTENTS OF TSSR FOR PRINTOUT.
004016' 017767  174262  174060                                                      MOV     @TSSR,ASTAT
004024'                         LET ACSR := @CSRA               ;SAVE CONTENTS OF TSDB(CSR) FOR PRINTOUT.
004024' 017767  174050  174050                                                      MOV     @CSRA,ACSR
                                .NLIST  MC
                                .LIST   ME
004032' 104403  000000' 005162'         MSGMS,BEGIN,FUNC        ;ASCII MESSAGE CALL WITH COMMON HEADER
004040' 104403  000000' 005206'         MSGMS,BEGIN,CMDA        ;ASCII MESSAGE CALL WITH COMMON HEADER
                                        ;*************************************************************
004046' 104405  000000' 000570'         ORDERS,BEGIN,TABLE      ;FUNCTION REJECTED
                                        ;*************************************************************
                                .LIST   MC
                                .NLIST  ME
004054'                         CALL DROPD                      ;CALL DROPD-DROP THE DEVICE.
004054' 004767  177156                                                              JSR     PC,DROPD
004060' 000207                  RTS PC                          ;RETURN
                        ;       ---------------------------------
```

```
                              ;*********************************************************************
                              ;*                                                                   *
                              ;*SUBROUTINE NAME:        TCC4                                       *
                              ;*                                                                   *
                              ;*FUNCTION:               PROCESSES TCC4 (RECOVERABLE ERROR).        *
                              ;*                                                                   *
                              ;*CALLING SEQUENCE:       CALL TCC4                                  *
                              ;*                                                                   *
                              ;*PARAMETERS PASSES:      NONE                                       *
                              ;*                                                                   *
                              ;*REGISTERS USED:         R2                                         *
                              ;*                                                                   *
                              ;*SUBORDINATE ROUTINES: RECER, UNREC                                 *
                              ;*                                                                   *
                              ;*ENTRY PREREQUISITES:    - COMMAND HAS BEEN ISSUED TO TS04.         *
                              ;*                        - INTERRUPT HAS BEEN RECEIVED.             *
                              ;*                                                                   *
                              ;*********************************************************************

004062'                       TCC4:  IF RTRYC LT RTRYL THEN          ;IF RETRY LIMIT HAS NOT BEEN REACHED:
004062' 026767 174232 174226                                                  CMP     RTRYC,RTRYL
004070' 002034                                                                BGE     50025$
004072'
004072' 004767 000210                  CALL RECER                     ;PRINT RECOVERABLE ERROR MESSAGE.
                                                                              JSR     PC,RECER
004076'
004076' 005267 174216                  LET RTRYC := RTRYC + #1        ;INCREMENT THE RETRY COUNT.
                                                                              INC     RTRYC
004102'
004102' 016777 174206 174440           LET @CMDPK1 := RCMD            ;MOVE RETRY CMD TO CMD PACKET.
                                                                              MOV     RCMD,@CMDPK1
004110'                                IF CMD EQ #RDR OR CMD EQ #RDF THEN  ;IF THE COMMAND IS A READ THEN:
004110' 026727 174176 100601                                                  CMP     CMD,#RDR
004116' 001404                                                                BEQ     50026$
004120' 026727 174166 100201                                                  CMP     CMD,#RDF
004126' 001011                                                                BNE     50027$
004130'                                                               50026$:
004130' 016702 174162                    LET R2 := RTRYL SHIFT -1       ;LET INDEX REGISTER EQUAL HALT THE RETRY LIM
                                                                              MOV     RTRYL,R2
004134' 006202                                                                ASR     R2
004136'
004136' 026702 174156                    IF RTRYC GT R2 THEN            ;IF RETRY COUNT IS MORE THAN
                                                                              CMP     RTRYC,R2
004142' 003403                                                                BLE     50030$
                                                                      ;HALF THE RETRY LIMIT THEN:
004144'
004144' 062777 020000 174376               LET @CMDPK1 := @CMDPK1 + #CM.OPP ;SET OPPOSITE DIRECTION BIT
                                                                              ADD     #CM.OPP,@CMDPK1
                                                                      ;IN CMD PACKET 1ST WORD.
004152'                                  ENDIF                        ;END RETRY DIRECTION CHECK.
004152'                                                               50030$:
004152'                                ENDIF                          ;END READ CMD CHECK.
004152'                                                               50027$:
004152' 112767 000001 174331           LET EREC :B= #1                ;SET ERROR RECOVERY REQUIRED FLAG.
                                                                              MOVB    #1,EREC
004160'                              ELSE                             ;ELSE-IF LIMIT NOT REACHED:
004160' 000402                                                                BR      50031$
004162'                                                               50025$:
004162' 004767 000204                  CALL UNRER                     ;PRINT UNRECOVERABLE ERROR MESSAGE.
                                                                              JSR     PC,UNRER
004166'                              ENDIF                            ;END RETRY LIMIT CHECK.
```



```
004166'                                                               50031$:
004166' 000207                       RTS PC                           ;RETURN
                              ;      --------------------------------
```

```
                        ;*****************************************************************************
                        ;*                                                                           *
                        ;*SUBROUTINE NAME:      TCC5                                                 *
                        ;*                                                                           *
                        ;*FUNCTION:             PROCESSES TCC5 (RECOVERABLE ERROR).                  *
                        ;*                                                                           *
                        ;*CALLING SEQUENCE:     CALL TCC5                                            *
                        ;*                                                                           *
                        ;*PARAMETERS PASSES:    NONE                                                 *
                        ;*                                                                           *
                        ;*REGISTERS USED:       NONE                                                 *
                        ;*                                                                           *
                        ;*SUBORDINATE ROUTINES: RECER, UNRER                                         *
                        ;*                                                                           *
                        ;*ENTRY PREREQUISITES:  - COMMAND HAS BEEN ISSUED TO TS04.                   *
                        ;*                      - INTERRUPT HAS BEEN RECEIVED.                       *
                        ;*                                                                           *
                        ;*****************************************************************************

004170'                 TCC5:   IF RTRYC LT RTRYL THEN          ;IF RETRY LIMIT HAS NOT BEEN REACHED THEN:
004170' 026767  174124  174120                                          CMP     RTRYC,RTRYL
004176' 002010                                                          BGE     50032$
004200'                           CALL RECER                      ;PRINT RECOVERABLE ERROR MESSAGE.
004200' 004767  000102                                                  JSR     PC,RECER
004204' 005267  174110            LET RTRYC := RTRYC + #1         ;INCREMENT THE RETRY COUNT.
004204'                                                                 INC     RTRYC
004210' 112767  000001  174273    LET EREC :8= #1                 ;SET ERROR RECOVERY REQUIRED FLAG.
004210'                                                                 MOVB    #1,EREC
004216' 000402                    ELSE                            ;ELSE-IF RETRY LIMIT REACHED:
004216'                                                                 BR      50033$
004220'                                                         50032$:
004220'                           CALL UNRER                      ;PRINT UNRECOVERABLE ERROR MESSAGE.
004220' 004767  000146                                                  JSR     PC,UNRER
004224'                           ENDIF                           ;END RETRY LIMIT CHECK.
004224'                                                         50033$:
004224' 000207                    RTS PC                          ;RETURN
                        ;       ---------------------------------
```



```
                        ;*****************************************************************************
                        ;*                                                                           *
                        ;*SUBROUTINE NAME:      TCC6                                                 *
                        ;*                                                                           *
                        ;*FUNCTION:             PROCESSES TCC6 (UNRECOVERABLE ERROR).                *
                        ;*                                                                           *
                        ;*CALLING SEQUENCE:     CALL TCC6                                            *
                        ;*                                                                           *
                        ;*PARAMETERS PASSES:    NONE                                                 *
                        ;*                                                                           *
                        ;*REGISTERS USED:       NONE                                                 *
                        ;*                                                                           *
                        ;*SUBORDINATE ROUTINES: NONE                                                 *
                        ;*                                                                           *
                        ;*ENTRY PREREQUISITES:  - COMMAND HAS BEEN ISSUED TO TS04.                   *
                        ;*                      - INTERRUPT HAS BEEN RECEIVED.                       *
                        ;*                                                                           *
                        ;*****************************************************************************

004226'                 TCC6:   CALL UNRER                      ;PRINT UNRECOVERABLE ERROR MESSAGE.
004226' 004767  000140                                                  JSR     PC,UNRER
004232' 000207                  RTS PC                          ;RETURN
                        ;       ---------------------------------
```

```
                        ;*********************************************************************
                        ;*                                                                   *
                        ;*SUBROUTINE NAME:      TCC7                                         *
                        ;*                                                                   *
                        ;*FUNCTION:             PROCESSES TCC7 (FATAL SUBSYSTEM ERROR).      *
                        ;*                                                                   *
                        ;*CALLING SEQUENCE:     CALL TCC7                                    *
                        ;*                                                                   *
                        ;*PARAMETERS PASSES:    NONE                                         *
                        ;*                                                                   *
                        ;*REGISTERS USED:       NONE                                         *
                        ;*                                                                   *
                        ;*SUBORDINATE ROUTINES: DROPD                                        *
                        ;*                                                                   *
                        ;*ENTRY PREREQUISITES:  - COMMAND HAS BEEN ISSUED TO TS04.           *
                        ;*                      - INTERRUPT HAS BEEN RECEIVED.               *
                        ;*                                                                   *
                        ;*********************************************************************

004234'                 TCC7:   LET ERRTYP := #55               ;LET ERROR TYPE EQUAL 55.
004234' 012767  000055  173644                                                  MOV     #55,ERRTYP
004242'                         LET ASTAT := @TSSR              ;SAVE CONTENTS OF TSSR FOR PRINTOUT.
004242' 017767  174036  173634                                                  MOV     @TSSR,ASTAT
004250'                         LET ACSR := @CSRA               ;SAVE CONTENTS OF TSDB(CSR) FOR PRINTOUT.
004250' 017767  173624  173624                                                  MOV     @CSRA,ACSR
                                .NLIST  MC
                                .LIST   ME
004256' 104403  000000' 005176'         MSGNS,BEGIN,FATA        ;ASCII MESSAGE CALL WITH COMMON HEADER
004264' 104403  000000' 005206'         MSGNS,BEGIN,CMDA        ;ASCII MESSAGE CALL WITH COMMON HEADER
                                        ;*************************************************************
004272' 104405  000000' 000570'         HDERS,BEGIN,TABLE       ;FATAL SUBSYSTEM ERROR
                                        ;*************************************************************
                                .LIST   MC
                                .NLIST  ME
004300'                         CALL DROPD                      ;CALL DROPD-DROP THE DEVICE.
004300' 004767  176732                                                          JSR     PC,DROPD
004304' 000207                  RTS PC                          ;RETURN
                        ;       ---------------------------------
```



```
                        ;*********************************************************************
                        ;*                                                                   *
                        ;*SUBROUTINE NAME:      RECER                                        *
                        ;*                                                                   *
                        ;*FUNCTION:             PRINT RECOVERABLE ERROR MESSAGE.             *
                        ;*                                                                   *
                        ;*CALLING SEQUENCE:     CALL RECER                                   *
                        ;*                                                                   *
                        ;*PARAMETERS PASSED:    NONE                                         *
                        ;*                                                                   *
                        ;*REGISTERS USED:       NONE                                         *
                        ;*                                                                   *
                        ;*SUBORDINATE ROUTINES: NONE                                         *
                        ;*                                                                   *
                        ;*********************************************************************

004306'                 RECER:  IF #BIT1 NOTSETIN SR1 THEN      ;IF BIT 1 NOT SET IN SR1 THEN:
004306' 032767  000002  173502                                                  BIT     #BIT1,SR1
004314' 001023                                                                  BNE     50034$
004316'                           LET ERRTYP := #53                ;LET ERROR TYPE EQUAL 53.
004316' 012767  000053  173562                                                  MOV     #53,ERRTYP
004324'                           LET ASTAT := @TSSR               ;SAVE CONTENTS OF TSSR FOR PRINTOUT.
004324' 017767  173754  173552                                                  MOV     @TSSR,ASTAT
004332'                           LET ACSR := @CSRA                ;SAVE CONTENTS OF TSDB(CSR) FOR PRINTOUT.
004332' 017767  173542  173542                                                  MOV     @CSRA,ACSR
                                  .NLIST NC
                                  .LIST ME
004340' 104403  000000' 005166'         MSGNS,BEGIN,RECO        ;ASCII MESSAGE CALL WITH COMMON HEADER
004346' 104403  000000' 005206'         MSGNS,BEGIN,CMDA        ;ASCII MESSAGE CALL WITH COMMON HEADER
                                        ;*************************************************************
004354' 104406  000000' 000570'         SOFERS,BEGIN,TABLE      ;RECOVERABLE ERROR
                                        ;*************************************************************
                                  .LIST NC
                                  .NLIST ME
004362'                         ELSE                            ;ELSE-IF BIT 1 SET IN SR1:
004362' 000402                                                                  BR      50035$
004364'                                                                 50034$:
004364'                           LET SOFCNT := SOFCNT + #1        ;UPDATE SOFT ERROR COUNT.
004364' 005267  173452                                                          INC     SOFCNT
004370'                         ENDIF                           ;END SWITCH REGISTER CHECK.
004370'                                                                 50035$:
004370' 000207                  RTS PC                          ;RETURN
                        ;       ---------------------------------
```

```
1548                            ;*********************************************************************
1549                            ;*                                                                   *
1550                            ;*SUBROUTINE NAME:      UNRER                                        *
1551                            ;*                                                                   *
1552                            ;*FUNCTIONS:            - PRINT RECOVERABLE ERROR MESSAGE.           *
1553                            ;*                      - DROP DEVICE IF BIT0 SET IN SR1.            *
1554                            ;*                                                                   *
1555                            ;*CALLING SEQUENCE:     CALL UNRER                                   *
1556                            ;*                                                                   *
1557                            ;*PARAMETERS PASSED:    NONE                                         *
1558                            ;*                                                                   *
1559                            ;*REGISTERS USED:       NONE                                         *
1560                            ;*                                                                   *
1561                            ;*SUBORDINATE ROUTINES: DROPD                                        *
1562                            ;*                                                                   *
1563                            ;*********************************************************************
1564
1565  004372'                   UNRER:  LET ERRTYP := #54               ;LET ERROR TYPE EQUAL 54.
1566  004372'  012767  000054  173506                                                  MOV     #54,ERRTYP
1567  004400'                          LET ASTAT := @TSSR               ;SAVE CONTENTS OF TSSR FOR PRINTOUT.
1568  004400'  017767  173700  173476                                                  MOV     @TSSR,ASTAT
1569  004406'                          LET ACSR := @CSRA                ;SAVE CONTENTS OF TSDB(CSR) FOR PRINTOUT.
1570  004406'  017767  173466  173466                                                  MOV     @CSRA,ACSR
1571                                   .NLIST  MC
1572                                   .LIST   ME
1573  004414'  104403  000000' 005172'       MSGMS,BEGIN,UNRE        ;ASCII MESSAGE CALL WITH COMMON HEADER
1574  004422'  104403  000000' 005206'       MSGMS,BEGIN,CMDA        ;ASCII MESSAGE CALL WITH COMMON HEADER
1575                                   ;*********************************************************************
1576  004430'  104405  000000' 000570'       ERDFS,BEGIN,TABLE       ;UNRECOVERABLE ERROR
1577                                   ;*********************************************************************
1578                                   .LIST   MC
1579                                   .NLIST  ME
1580  004436'                          IF #BIT0 SETIN SR1 THEN          ;IF BIT0 SET IN SR1 THEN:
1581  004436'  032767  000001  173352                                                  BIT     #BIT0,SR1
1582  004444'  001402                                                                  BEQ     50036$
1583  004446'                            CALL DROPD                       ;CALL DROPD-DROP THE DEVICE.
1584  004446'  004767  176564                                                          JSR     PC,DROPD
1585  004452'                          ENDIF                            ;END SWITCH REGISTER CHECK.
1586  004452'                                                                        50036$:
1587  004452'  000207                 RTS PC                           ;RETURN
1588                            ;       ---------------------------------
1589
```



```
1590                            ;*********************************************************************
1591                            ;*                                                                   *
1592                            ;*              ROUTINES TO SERVICE UNEXPECTED INTERRUPTS            *
1593                            ;*                                                                   *
1594                            ;*********************************************************************
1595
1596  004454'                   UID0:   LET UIFL0 :B= #1                 ;SET DEVICE 0 UNEXPECTED INTERRUPT FLAG.
1597  004454'  112767  000001  174030                                                  MOVB    #1,UIFL0
1598  004462'  000002                 RTI                              ;RETURN FROM INTERRUPT.
1599                            ;       ---------------------------------
1600
1601  004464'                   UID1:   LET UIFL1 :B= #1                 ;SET DEVICE 1 UNEXPECTED INTERRUPT FLAG.
1602  004464'  112767  000001  174021                                                  MOVB    #1,UIFL1
1603  004472'  000002                 RTI                              ;RETURN FROM INTERRUPT.
1604                            ;       ---------------------------------
1605
1606  004474'                   UID2:   LET UIFL2 :B= #1                 ;SET DEVICE 2 UNEXPECTED INTERRUPT FLAG.
1607  004474'  112767  000001  174012                                                  MOVB    #1,UIFL2
1608  004502'  000002                 RTI                              ;RETURN FROM INTERRUPT.
1609                            ;       ---------------------------------
1610
1611  004504'                   UID3:   LET UIFL3 :B= #1                 ;SET DEVICE 3 UNEXPECTED INTERRUPT FLAG.
1612  004504'  112767  000001  174003                                                  MOVB    #1,UIFL3
1613  004512'  000002                 RTI                              ;RETURN FROM INTERRUPT.
1614                            ;       ---------------------------------
1615
```

```
 1616
 1617                                           .NLIST  BEX
      004514' 047503  052116  047522  MSG1:    .ASCIZ  "CONTROLLER NOT READY"
      004541'   122  040505  054504  MSG2:    .ASCIZ  "READY DID NOT DROP"
      004564' 047125  054105  042520  MSG3:    .ASCIZ  "UNEXPECTED INTERRUPT"
      004611'   123  042520  044503  MSG4:    .ASCIZ  "SPECIAL CONDITION STATUS"
      004642' 042504  044526  042503  MSG5:    .ASCIZ  "DEVICE OFF LINE"
      004662' 040524  042520  051440  MSG6:    .ASCIZ  "TAPE STATUS ALERT"
      004704' 052506  041516  044524  MSG7:    .ASCIZ  "FUNCTION REJECTED"
      004726' 042522  047503  042525  MSG8:    .ASCIZ  "RECOVERABLE ERROR"
      004750' 047125  042522  047503  MSG9:    .ASCIZ  "UNRECOVERABLE ERROR"
      004774' 040506  040524  020114  MSG10:   .ASCIZ  "FATAL SUBSYSTEM ERROR"
      005022' 051104  050117  042520  MSG11:   .ASCIZ  "DROPPED DEVICE"
      005042' 041440  046517  040515  MSG12:   .ASCIZ  " COMMAND"
      005053'   123  052105  041440  SCHAS:   .ASCIZ  "SET CHAR"
      005064' 051105  051501  000105  ERSAS:   .ASCIZ  "ERASE"
      005072' 051127  052111  000105  WRTAS:   .ASCIZ  "WRITE"
      005100' 042522  042101  051040  RDRAS:   .ASCIZ  "READ REV"
      005111'   122  040505  020104  RDFAS:   .ASCIZ  "READ FWD"
      005122' 042522  044527  042116  RWDAS:   .ASCIZ  "REWIND"
                                               .LIST   BEX
 1618           005132'                          .EVEN
 1619
 1620 005132' 004514'                  CONT:    MSG1
 1621 005134' 177777                            TERM
 1622 005136' 004541'                  READ:    MSG2
 1623 005140' 177777                            TERM
 1624 005142' 004564'                  UNEX:    MSG3
 1625 005144' 177777                            TERM
 1626 005146' 004611'                  SPEC:    MSG4
 1627 005150' 177777                            TERM
 1628 005152' 004642'                  DEVI:    MSG5
 1629 005154' 177777                            TERM
 1630 005156' 004662'                  TAPE:    MSG6
 1631 005160' 177777                            TERM
 1632 005162' 004704'                  FUNC:    MSG7
 1633 005164' 177777                            TERM
 1634 005166' 004726'                  RECO:    MSG8
 1635 005170' 177777                            TERM
 1636 005172' 004750'                  UNRE:    MSG9
 1637 005174' 177777                            TERM
 1638 005176' 004774'                  FATA:    MSG10
 1639 005200' 177777                            TERM
 1640 005202' 005022'                  DROE:    MSG11
 1641 005204' 177777                            TERM
 1642 005206' 005053'                  CMDA:    SCHAS                  ;ADR OF CMD ASCII WILL BE STORED HERE.
 1643 005210' 005042'                           MSG12
 1644 005212' 177777                            TERM
 1645
 1646           000001                          .END
```



```
ACSR     000102R          246#     958*   1103*   1136*   1209*   1247*   1289*   1328*   1489*   1528*   1570*
ADDR     000006R          212#     545
ADDR22=  001000           275#
ASB      000106R          250#
ASSFMB=  000010           281
ASTAT    000104R          248#     956*   1101*   1134*   1207*   1245*   1287*   1326*   1487*   1526*   1568*
AWAS     000110R          251#
BCON     000560R          385      386     387     413     414     415     465#    1009
BEGIN    000000R          209#     595     596     606     633     634     693     720     735     762     770     777     781
                          961      962     964     1050    1091    1092    1106    1107    1109    1139    1140    1142    1153
                         1156     1212    1213    1215    1250    1251    1253    1292    1293    1295    1331    1332    1334
                         1492     1493    1495    1531    1532    1534    1573    1574    1576
BIT0   = 000001          275#     600     1521
BIT1   = 000002          275#     600     1521
BIT10  = 002000          275#
BIT11  = 004000          275#
BIT12  = 010000          275#
BIT13  = 020000          275#
BIT14  = 040000          275#
BIT15  = 100000          275#
BIT2   = 000004          275#
BIT3   = 000010          275#
BIT4   = 000020          275#
BIT5   = 000040          275#
BIT6   = 000100          275#
BIT7   = 000200          275#
BIT8   = 000400          275#
BIT9   = 001000          275#
BREAK$=  104407          275#     1091    1092
BR1      000012R         214#     573
BR2      000013R         215#
BTOD$  = 104421          275#
BUFIN    000062R         257#     518#
CBUFEA   000256R         325#     612
CBUFPA   000254R         324#     600     618
CBUFVA   000252R         323#     596     603*    606     621     623     626     629
CDATA$=  104412          275#     720     735
CDERCT   000144R         265#
CDWDCT   000146R         266#
CMD      000312R         348#     822*    1372    1374
CMDA     005206R         828*     962     1107    1140    1213    1251    1293    1332    1493    1532    1574    1642#
CMDADR   000546R         456#     612*    613*    614*    615*    616*    618*    1126
CMDPKT   000622R         323      492#
CMDPK1   000550R         458#     621#    820*    1370*   1385*
CMDPK2   000552R         459#     623*    624*    830*
CMDPK3   000554R         460#     626*    627*    832*    833*    834*    835*    836*
CMDPK4   000556R         461#     629*    630*    839*    840*
CM.OPP=  020000          300#     1385
CONFIG   000056R         234#
CONT     005132R         1106     1620#
CONTUE   002074R         581      594#
CRETRY=  000010          315#     376     383     411
CSRA     000100R         244#     870*    958     1103    1126*   1136    1209    1247    1289    1328    1489    1528    1570
CVSET    002626R         656      664     702     709     724     742     813#
DATCK$=  104411          275#
DATER$=  104404          275#
```

```
DCNT    000274R          335#    473     538     918*    949     977*    979     993*    1004
DEVI    005152R         1250    1628#
DIAGMC= 000000           281
DIND    000300R          337#    528*    1014    1045*
DINX    000276R          336#    540     697     747     868     920*    995*    1004*   1005*   1007    1039    1161
DMOD    000510R          421#    530*    685     985*
DROE    005202R         1050    1640#
DROP    000507R          420#    662     707     714     729     926*    973*    1047*
DROPD   003236R          969    1036#    1114    1147    1220    1258    1339    1500    1584
DVID1   000014R          216#    528
DVSET   002726R          567     655     701     865#
ENDIT$= 104413           275#    762
END$    = 104410         275#    770     777     781
EOD     000506R          419#    535     652     878*    924*    982     991*
ERASE   000400R          381#    665
EREC    000511R          422#   1084*    1174    1392*   1430*
ERRTYP  000106R          249#    717*    732*    954*    1099*   1132*   1205*   1243*   1285*   1324*   1485*   1524*   1566*
ERS     = 100611         289#    381     382
ERSAS   005064R          384    1617#
EXIT$ = 104400           275#   1153
FATA    005176R         1492    1638#
FIN1    = 050010         773#
FINIT   002616R          688     774#
FREE    000150R          267#
FUNC    005162R         1331    1632#
GETPAS= 104415           275#    595     596     606     633     634
GWBUF$= 104414           275#    693
HRDCNT  000044R          229#
HRDER$= 104405           275#    964     1109    1142    1253    1334    1495    1576
HRDPAS  000050R          231#
ICCRTN  003604R         1115    1148    1179#
ICCS    003304R          659     667     705     712     727     745     1081#
ICCSR   003306R         1083#   1176
ICCS1   003530R         1124    1154#
ICCS2   003536R         1156    1160#
ICONT   000036R          226#    681
ICOUNT  000040R          227#    681
IDNUM   000122R          256#
IDP     002774R          532     649     917#
IMODX.= 000000           268#    694
INIT    000030R          223#
INTR    000120R          255#    526*
MAIN    = 050000         679#
MAP22$= 104416           275#
MBUFEA  000264R          328#    642
MBUFPA  000262R          327#    639
MBUFVA  000260R          326#    633
MODNAM  000000R          210#
MODSP   000252R          324     273#
MSGBLK  000634R          329     501#    639*    642*    643*    644*    645*    646*
MSGCNT= 000016           316#    503
MSGN$ = 104403           275#    961     962     1050    1106    1107    1139    1140    1212    1213    1250    1251    1292
                        1293    1331    1332    1492    1493    1531    1532    1573    1574
MSGPKT  000644R          326     474     475     476     477     478     479     480     508#
MSGS$ = 104402           275#
MSG$  = 104401           275#
```



```
MSG1    004514R         1617#   1620
MSG10   004774R         1617#   1638
MSG11   005022R         1617#   1640
MSG12   005042R         1617#   1643
MSG2    004541R         1617#   1622
MSG3    004564R         1617#   1624
MSG4    004611R         1617#   1626
MSG5    004642R         1617#   1628
MSG6    004662R         1617#   1630
MSG7    004704R         1617#   1632
MSG8    004726R         1617#   1634
MSG9    004752R         1617#   1636
NULL    = 000000         275#
OPEN    = 000000         211     217     218     219     220     237     238     239     240     241     242     243     244
                         246     248     250     251     253     254     255     258     259     261     262     264     265
                         266     267     275#
OTOAS = 104420           275#
PASCNT  000034R          225#    585
PIRQS = 000004           275#   1156
POPSP = 005726           275#
POPSP2= 022626           275#
PRTY  = 000000           275#
PRTY0 = 000000           215     275#
PRTY1 = 000040           275#
PRTY2 = 000100           275#
PRTY3 = 000140           275#
PRTY4 = 000200           275#
PRTY5 = 000240           214     275#
PRTY6 = 000300           275#
PRTY7 = 000340           275#
PS    = 177776           275#
PSW   = 177776           275#
PUSH  = 005746           275#
PUSH2 = 024646           275#
RAND$ = 104417           275#
RANNUM  000054R          233#
RBUFEA  000130R          259#    400     407
RBUFPA  000126R          258#    399     406     720     735
RBUFSZ  000132R          260#    401     408     521
RBUFVA  000124R          257#    595
RCMD    000314R          349#    824*    1370
RDF   = 100201           292#    402     1374
RDFAS   005111R          405    1617#
RDR   = 100601           291#    395     1372
RDRAS   005100R          398    1617#
READ    005136R         1139    1622#
RECER   004306R         1426    1520#
RECO    005116R         1531    1634#
RECORD  000310R          344#    481     876*
RESTRT  002064R          252     584#
RES1    000056R          235#
RES2    000060R          236#
REWIND  000470R          409#    743
RFC     000650R          510#
RFR   = 101201           296#    403
RFWD    000452R          402#    725
```

```
RRETRY= 000020       314#    397     404
RREV    000434R      395#    710
RRR   = 101601       295#    396
RSTRT   000112R      252#
RTRYC   000320R      351#    482     843*    1363    1368*   1381    1423    1428*
RTRYL   000316R      350#    826*    1363    1378    1423
RWD   = 102210       293#    409     410
RWDAS   005122R      412     1617#
SBADR   000102R      245#
SBUFEA  000272R      331#    379
SBUFPA  000270R      330#    378
SBUFSZ  000616R      380     487#
SBUFVA  000266R      329#    634
SCDFLT= 000040       318#    504
SCH   = 140204       288#    374     375
SCHAR   000362R      374#    657
SCHAS   005053R      377     1617#   1642
SCHCNT= 000004       317#    487
SOFCNT  000042R      228#    1542*
SOPER$= 104406       275#    1215    1295    1534
SOFPAS  000046R      230#
SPEC    005146R     1212     1626#
SPOINT  000032R      224#
SPSIZ = 000040         1#    268
SR1     000016R      217#    559     1521    1581
SR2     000020R      218#
SR3     000022R      219#
SR4     000024R      220#
START   001662R      223     520#    587
STAT    000026R      222#
SVR0    000062R      237#
SVR1    000064R      238#
SVR2    000066R      239#
SVR3    000070R      240#
SVR4    000072R      241#
SVR5    000074R      242#
SVR6    000076R      243#
SYSCNT  000052R      232#
TABLE   000570R      473#    964     1109    1142    1215    1253    1295    1334    1495    1534    1576
TAPE    005156R     1292     1630#
TCCRA   000516R      436#    1170
TCC0    003610R      436     1204#
TCC1    003660R      437     1242#
TCC2    003732R      438     1281#
TCC3    004010R      439     1323#
TCC4    004062R      440     1362#
TCC5    004170R      441     1422#
TCC6    004226R      442     1460#
TCC7    004234R      443     1484#
TERM  = 177777       319#    483     1621    1623    1625    1627    1629    1631    1633    1635    1637    1639    1641
                    1644
TIME    000620R      488#    1086*   1096    1119*
TREC    000352R      368#    563*    699*    749*    876
TRPDFD= 000022       275#
TSSR    000304R      342#    575*    872*    956     1043*   1101    1121    1129    1134    1165    1168    1207    1245
                    1287     1326    1487    1526    1568
```



```
TS.SC = 100000       305#    1165
TS.SSR= 000200       304#    1121    1129
TTSDB   000322R      356#    545*    546*    548     870
TTSSR   000332R      360#    548*    549*    872
TVECT   000342R      364#    554*    559*    874
UIADR   000536R      448#    571     1041    1163
UID0    004454R      448     1596#
UID1    004464R      449     1601#
UID2    004474R      450     1606#
UID3    004504R      451     1611#
UIFL0   000512R      427#    565*    951     1597*
UIFL1   000513R      428#    1602*
UIFL2   000514R      429#    1607*
UIFL3   000515R      430#    1612*
UNEX    005142R      961     1624#
UNRE    005172R     1573     1636#
UNRER   004372R     1397     1435    1461    1565#
UPDP    003024R      577     672     759     946#
UPDRTN  003232R      986     1016#
VDIND   000302R      338#    922*    997*    1009*   1014    1045
VECT    000306R      343#    569     874*    1041*   1124*   1163*
VECTOR  000010R      213#    554
WASADR  000104R      247#
WBUFEA  000136R      262#    393
WBUFPA  000134R      261#    392
WBUFRQ  000140R      263#
WBUFSZ  000142R      264#    394     524
WDFR    000116R      254#    524*
WDTO    000114R      253#    521*    522*
WRETRY= 000020       313#    390
WRITE   000416R      388#    703
WRT   = 100205       290#    388
WRTAS   005072R      391     1617#
WTR   = 101205       294#    389
XFLAG   000005R      211#
XSTAT0  000652R      511#    740     1282
XSTAT1  000654R      512#
XSTAT2  000656R      513#
XSTAT3  000660R      514#
XSO.EO= 000001       309#    740     1282
$BGNLE= 177777       281#
$ERFLG= 000400       281#    521#    522#    523     524#    525     526#    527     528#    529     530#    531     532#
                     538#    539     540#    541     542#    543#    544     545#    546#    547     548#    549#    550
                     554#    555     559#    560     563#    564     565#    566     567#    569#    570     571#    572
                     573#    574     575#    576     577#    603#    604     612#    613#    617     618#    619     621#
                     622     623#    624#    625     626#    627#    628     629#    630#    631     639#    640     642#
                     643#    647     649#    655#    659#    667#    672#    697#    698     699#    700     701#    705#
                     712#    717#    718     727#    732#    733     745#    747#    748     749#    750     759#    816#
                     817     820#    821     822#    823     824#    825     826#    827     828#    829     830#    831
                     832#    833#    837     839#    840#    841     843#    844     868#    869     870#    871     872#
                     873     874#    875     876#    877     878#    879     918#    919     920#    921     922#    923
                     924#    925     926#    927     949#    950     954#    955     956#    957     958#    959     969#
                     973#    974     977#    978     985#    986     991#    992     993#    994     995#    996     997#
                     998     1004#   1005#   1006    1007#   1008    1009#   1010    1039#   1040    1041#   1042    1043#
                    1044     1045#   1046    1047#   1048    1084#   1085    1086#   1087    1099#   1100    1101#   1102
                    1103#    1104    1114#   1119#   1120    1124#   1125    1126#   1127    1132#   1133    1134#   1135
```

```
                 1136#  1137   1147#  1161#  1162   1163#  1164   1168#  1169#  1170   1205#  1206   1207#
                 1208   1209#  1210   1220#  1243#  1244   1245#  1246   1247#  1248   1258#  1285#  1286
                 1287#  1288   1289#  1290   1324#  1325   1326#  1327   1328#  1329   1339#  1366#  1368#
                 1369   1370#  1371   1378#  1379#  1380   1385#  1386   1392#  1393   1397#  1426#  1428#
                 1429   1430#  1431   1435#  1461#  1485#  1486   1487#  1488   1489#  1490   1500#  1524#
                 1525   1526#  1527   1528#  1529   1542#  1543   1566#  1567   1568#  1569   1570#  1571
                 1584#  1597#  1598   1602#  1603   1607#  1608   1612#  1613
$F$AND= 000310    281#   537    553    587    602    654    664    683    687    709    716    731    742
                  953    981    984   1098   1131   1167   1176   1284   1365   1374   1383   1425   1523
                 1583
$F$BAD= 000401    281#   521    524    526    528    530    537    538    540    542    545    548    553
                  554    559    563    565    569    571    573    575    587    602    603    612    618
                  621    623    626    629    639    642    654    664    683    687    697    699    709
                  716    717    731    732    742    747    749    816    820    822    824    826    828
                  830    832    839    843    868    870    872    874    876    878    918    920    922
                  924    926    949    953    954    956    958    973    977    981    984    985    991
                  993    995    997   1004   1007   1009   1039   1041   1043   1045   1047   1084   1086
                 1098   1099   1101   1103   1119   1124   1126   1131   1132   1134   1136   1161   1163
                 1167   1168   1176   1205   1207   1209   1243   1245   1247   1284   1285   1287   1289
                 1324   1326   1328   1365   1368   1370   1377   1378   1383   1385   1392   1425   1428
                 1430   1485   1487   1489   1523   1524   1526   1528   1542   1566   1568   1570   1583
                 1597   1602   1607   1612
$F$BLA= 000170    281#
$F$CAS= 000150    281#
$F$DEC= 000220    281#
$F$DO = 000340    281#   537    654    683
$F$FAL= 000405    281#
$F$GOO= 000400    281#   521    522    523    524    525    526    527    528    529    530    531    532
                  534    535    538    539    540    541    542    543    544    545    546    547    548
                  549    550    551    554    555    559    560    563    564    565    566    567    569
                  570    571    572    573    574    575    576    577    585    600    603    604    612
                  613    617    618    619    621    622    623    624    625    626    627    628    629
                  630    631    639    640    642    643    647    649    651    652    655    659    662
                  667    672    680    681    685    697    698    699    700    701    705    707    712
                  714    717    718    727    729    732    733    740    745    747    748    749    750
                  759    816    817    820    821    822    823    824    825    826    827    826    829
                  830    831    832    833    837    839    840    841    843    844    868    869    870
                  871    872    873    874    875    876    877    878    879    918    919    920    921
                  922    923    924    925    926    927    949    950    951    954    955    956    957
                  958    959    969    973    974    977    978    979    982    985    986    991    992
                  993    994    995    996    997    998   1004   1005   1006   1007   1008   1009   1010
                 1039   1040   1041   1042   1043   1044   1045   1046   1047   1048   1084   1085   1086
                 1087   1096   1099   1100   1101   1102   1103   1104   1114   1119   1120   1124   1125
                 1126   1127   1129   1132   1133   1134   1135   1136   1137   1147   1161   1162   1163
                 1164   1165   1168   1169   1170   1174   1205   1206   1207   1208   1209   1210   1220
                 1243   1244   1245   1246   1247   1248   1258   1282   1285   1286   1287   1288   1289
                 1290   1324   1325   1326   1327   1328   1329   1339   1363   1366   1368   1369   1370
                 1371   1372   1378   1379   1380   1381   1385   1386   1392   1393   1397   1423   1426
                 1428   1429   1430   1431   1435   1461   1485   1486   1487   1488   1489   1490   1500
                 1521   1524   1525   1526   1527   1528   1529   1542   1543   1566   1567   1568   1569
                 1570   1571   1581   1584   1597   1598   1602   1603   1607   1608   1612   1613
$F$IF = 000110    281#   551    556    561    585    589    600    610    662    670    685    690    707
                  714    729    740    751    753    755    757    951    971    979    982    988    999
                 1001   1012   1096   1117   1129   1150   1165   1172   1174   1178   1282   1300   1363
                 1372   1374   1381   1388   1390   1394   1399   1423   1432   1437   1521   1539   1544
                 1581   1586
```



```
$F$INC= 000210    281#
$F$LOO= 000200    281#
$F$NAM= 000160    281#   679    773
$F$NO = 000403    281#   521    523    524    525    526    527    528    529    530    531    538    539
                  540    541    542    544    545    547    548    550    551    554    555    559    560
                  563    564    565    566    569    570    571    572    573    574    575    576    585
                  600    603    604    612    617    618    619    621    622    623    625    626    628
                  629    631    639    640    642    647    681    697    698    699    700    717    718
                  732    733    740    747    748    749    750    816    817    820    821    822    823
                  824    825    826    827    828    829    830    831    832    837    839    841    843
                  844    868    869    870    871    872    873    874    875    876    877    878    879
                  918    919    920    921    922    923    924    925    926    927    949    950    954
                  955    956    957    958    959    973    974    977    978    979    985    986    991
                  992    993    994    995    996    997    998   1004   1006   1007   1008   1009   1010
                 1039   1040   1041   1042   1043   1044   1045   1046   1047   1048   1084   1085   1086
                 1087   1096   1099   1100   1101   1102   1103   1104   1119   1120   1124   1125   1126
                 1127   1129   1132   1133   1134   1135   1136   1137   1161   1162   1163   1164   1165
                 1168   1170   1205   1206   1207   1208   1209   1210   1243   1244   1245   1246   1247
                 1248   1282   1285   1286   1287   1288   1289   1290   1324   1325   1326   1327   1328
                 1329   1363   1368   1369   1370   1371   1372   1374   1378   1380   1381   1385   1386
                 1392   1393   1423   1428   1429   1430   1431   1485   1486   1487   1488   1489   1490
                 1521   1524   1525   1526   1527   1528   1529   1542   1543   1566   1567   1568   1569
                 1570   1571   1581   1597   1598   1602   1603   1607   1608   1612   1613
$F$OR = 000320    281#   537    553    587    602    654    664    683    687    709    716    731    742
                  953    981    984   1098   1131   1167   1176   1284   1365   1374   1383   1425   1523
                 1583
$F$RTI= 000350    281#
$F$RTN= 000300    281#
$F$SEL= 000140    281#
$F$THE= 000330    281#   553    587    602    664    687    709    716    731    742    953    981    984
                 1098   1131   1167   1176   1284   1365   1377   1383   1425   1523   1583
$F$TRU= 000404    281#
$F$UNT= 000130    281#   975   1014   1088   1121
$F$WHI= 000120    281#   534    535    579    651    652    674    680    681    767   1374
$F$YES= 000402    281#   523    525    527    529    531    535    539    541    544    547    550    551
                  555    556    560    561    564    566    570    572    574    576    585    589    600
                  604    610    617    619    622    625    628    631    640    647    652    662    670
                  681    685    690    698    700    707    714    718    729    733    740    748    750
                  751    753    755    757    817    821    823    825    827    829    831    837    841
                  844    869    871    873    875    877    879    919    921    923    925    927    950
                  951    955    957    959    971    974    978    979    982    986    988    992    994
                  996    998    999   1001   1006   1008   1010   1012   1040   1042   1044   1046   1048
                 1085   1087   1096   1100   1102   1104   1117   1120   1125   1127   1129   1133   1135
                 1137   1150   1162   1164   1165   1170   1172   1174   1178   1206   1208   1210   1244
                 1246   1248   1282   1286   1288   1290   1300   1325   1327   1329   1363   1369   1371
                 1372   1374   1380   1381   1386   1388   1390   1393   1394   1399   1423   1429   1431
                 1432   1437   1486   1488   1490   1521   1525   1527   1529   1539   1543   1544   1567
                 1569   1571   1581   1586   1598   1603   1608   1613
$IFLEV= 177777    281#   551#   561#   585#   589#   600#   610#   662#   670#   685#   690#   707#   714#
                  729#   740#   751#   753#   755#   757#   951#   971#   979#   982#   999#  1012#  1096#
                 1117#  1129#  1150#  1165#  1172#  1174#  1178#  1282#  1300#  1363#  1372#  1381#  1388#
                 1390#  1399#  1423#  1437#  1521#  1544#  1581#  1586#
$ISK0 = 000001    551#   561    585#   589    600#   610    662#   670    685#   690    707#   757    951#
                  971    979#  1012   1096#  1117   1129#  1150   1165#  1172   1174#  1178   1282#  1300
                 1363#  1399   1423#  1437   1521#  1544   1581#  1586
$ISK1 = 000001    714#   755    982#   999   1372#  1390
```

```
$ISK2 = 000001     729#   753   1381#  1388
$ISK3 = 000001     740#   751
$LOCTA= 177777     281#   534    536    552    556    557    561    579    580    586    589    601    610
                   651    653    663    670    674    675    679    680    682    686    690    708    715
                   730    741    751    753    755    757    767    768    952    971    975    980    983
                   988    989    999   1001   1002   1012   1015   1088   1097   1117   1122   1130   1150
                  1166   1172   1175   1178   1283   1300   1364   1373   1375   1376   1382   1388   1390
                  1394   1395   1399   1424   1432   1433   1437   1522   1539   1540   1544   1582   1586
$LSTIN= 000001     281#   521    522    524    526    528    530    532    535    536    538    540    542
                   543    545    546    548    549    551    552    554    556    559    563    565    567
                   569    571    573    575    577    579    585    586    600    601    603    612    613
                   614    615    616    618    621    623    624    626    627    629    630    639    642
                   643    644    645    646    649    652    653    655    659    662    663    667    672
                   674    681    682    685    686    688    697    699    701    705    707    708    712
                   714    715    717    727    729    730    732    740    741    745    747    749    759
                   767    774    814    816    820    822    824    826    828    830    832    833    834
                   835    836    839    840    843    845    866    868    870    872    874    876    878
                   880    918    920    922    924    926    947    949    951    952    954    956    958
                   969    973    977    979    980    982    983    985    988    991    993    995    997
                  1001   1004   1005   1007   1009   1014   1015   1017   1037   1039   1041   1043   1045
                  1047   1054   1082   1084   1086   1096   1097   1099   1101   1103   1114   1119   1121
                  1122   1124   1126   1129   1130   1132   1134   1136   1147   1161   1163   1165   1166
                  1168   1169   1174   1175   1180   1205   1207   1209   1220   1243   1245   1247   1258
                  1282   1283   1285   1287   1289   1324   1326   1328   1339   1363   1364   1366   1368
                  1370   1372   1373   1374   1375   1378   1379   1381   1382   1385   1392   1394   1397
                  1423   1424   1426   1428   1430   1432   1435   1461   1485   1487   1489   1500   1521
                  1522   1524   1526   1528   1539   1542   1566   1568   1570   1581   1582   1584   1597
                  1602   1607   1612
$LSTTA= 000001     281#   282#   534    557    561    580    589    610    651    670    675    678    680
                   690    751    753    755    757    768    971    975    989    999   1002   1012   1088
                  1117   1150   1172   1178   1300   1376   1388   1390   1395   1399   1433   1437   1540
                  1544   1586
$MAIN   002370R    678#
$NESTL= 000001     281#   534#   551#   556    561#   579#   585#   589#   600#   610#   651#   662#   670#
                   674#   679#   680#   685#   690#   707#   714#   729#   740#   751#   753#   755#   757#
                   767#   773#   951#   971#   975#   979#   982#   988    999#  1001   1012#  1014#  1088#
                  1096#  1117#  1121#  1129#  1150#  1165#  1172#  1174#  1178#  1282#  1300#  1363#  1372#
                  1381#  1388#  1390#  1394   1399#  1423#  1432   1437#  1521#  1539   1544#  1581#  1586#
$NSK0 = 000160     534#   579    585#   589    600#   610    651#   674    679#
$NSK1 = 000160     551#   556    561    662#   670    680#   767    773#
$NSK2 = 000110     685#   690    707#   757    951#   971    975#  1014   1088#  1121   1129#  1150   1165#
                  1172   1174#  1178   1282#  1300   1363#  1394   1399   1423#  1432   1437   1521#  1539
                  1544   1581#  1586
$NSK3 = 000110     714#   755    979#  1001   1012   1096#  1117   1372#  1390
$NSK4 = 000110     729#   753    982#   988    999   1381#  1388
$NSK5 = 000110     740#   751
$SAVLE= 177777     281#   579#   580#   674#   675#   767#   768#
$SSK0 = 050002     579#   580    674#   675    767#   768
$TAGLE= 177777     281#   535#   537#   553#   557#   558#   561#   579#   587#   589#   602#   610#   652#
                   654#   664#   670#   674#   681#   683#   687#   690#   709#   716#   731#   742#   751#
                   753#   755#   757#   767#   953#   971#   976#   981#   984#   989#   990#   999#  1002#
                  1003#  1012#  1014#  1089#  1098#  1117#  1121#  1131#  1150#  1167#  1172#  1176#  1178#
                  1284#  1300#  1365#  1377#  1383#  1388#  1390#  1395#  1396#  1399#  1425#  1433#  1434#
                  1437#  1523#  1540#  1541#  1544#  1583#  1586#
$TAGNU= 050037     281#   534    535#   536    537#   552    553#   556    558#   586    587#   601    602#
                   651    652#   653    654#   663    664#   679#   680    681#   682    683#   686    687#
```



```
                   708    709#   715    716#   730    731#   741    742#   773#   952    953#   975    976#
                   980    981#   983    984#   988    990#  1001   1003#  1088   1089#  1097   1098#  1130
                  1131#  1166   1167#  1175   1176#  1283   1284#  1364   1365#  1373   1375   1376   1377#
                  1382   1383#  1394   1396#  1424   1425#  1432   1434#  1522   1523#  1539   1541#  1582
                  1583#
$TEMP = 000402     521#   523#   524#   525#   526#   527#   528#   529#   530#   531#   538#   539#   540#
                   541#   542#   544#   545#   547#   548#   550#   554#   555#   556#   557#   559#   560#
                   561#   563#   564#   565#   566#   569#   570#   571#   572#   573#   574#   575#   576#
                   579#   580#   589#   603#   604#   610#   612#   617#   618#   619#   621#   622#   623#
                   625#   626#   628#   629#   631#   639#   640#   642#   647#   670#   674#   675#   690#
                   697#   698#   699#   700#   717#   718#   732#   733#   747#   748#   749#   750#   751#
                   753#   755#   757#   767#   768#   816#   817#   820#   821#   822#   823#   824#   825#
                   826#   827#   828#   829#   830#   831#   832#   837#   839#   841#   843#   844#   868#
                   869#   870#   871#   872#   873#   874#   875#   876#   877#   878#   879#   918#   919#
                   920#   921#   922#   923#   924#   925#   926#   927#   949#   950#   954#   955#   956#
                   957#   958#   959#   971#   973#   974#   977#   978#   985#   986#   988#   989#   991#
                   992#   993#   994#   995#   996#   997#   998#   999#  1001#  1002#  1004#  1006#  1007#
                  1008#  1009#  1010#  1012#  1014#  1015   1039#  1040#  1041#  1042#  1043#  1044#  1045#
                  1046#  1047#  1048#  1084#  1085#  1086#  1087#  1099#  1100#  1101#  1102#  1103#  1104#
                  1117#  1119#  1120#  1121#  1122   1124#  1125#  1126#  1127#  1132#  1133#  1134#  1135#
                  1136#  1137#  1150#  1161#  1162#  1163#  1164#  1168#  1170#  1172#  1178#  1205#  1206#
                  1207#  1208#  1209#  1210#  1243#  1244#  1245#  1246#  1247#  1248#  1285#  1286#  1287#
                  1288#  1289#  1290#  1300#  1324#  1325#  1326#  1327#  1328#  1329#  1368#  1369#  1370#
                  1371#  1378#  1380#  1385#  1386#  1388#  1390#  1392#  1393#  1394#  1395#  1399#  1428#
                  1429#  1430#  1431#  1432#  1433#  1437#  1485#  1486#  1487#  1488#  1489#  1490#  1524#
                  1525#  1526#  1527#  1528#  1529#  1539#  1540#  1542#  1543#  1544#  1566#  1567#  1568#
                  1569#  1570#  1571#  1586#  1597#  1598#  1602#  1603#  1607#  1608#  1612#  1613#
$TSK0 = 050036     535#   579    587#   589    602#   610    652#   674    681#   767    953#   971    976#
                  1014   1089#  1121   1131#  1150   1167#  1172   1176#  1178   1284#  1300   1365#  1395
                  1396#  1399   1425#  1433   1434#  1437   1523#  1540   1541#  1544   1583#  1586
$TSK1 = 050027     537#   579    654#   674    683#   767    981#  1002   1003#  1012   1098#  1117   1377#
                  1390
$TSK2 = 050030     553#   557    558#   561    664#   670    687#   690    709#   757    984#   989    990#
                   999   1383#  1388
$TSK3 = 050005     716#   755
$TSK4 = 050006     731#   753
$TSK5 = 050007     742#   751
$$ARGC= 000000     281#
$$BYTE= 000403     281#   535#   551#   585#   600#   652#   662#   681#   685#   707#   714#   729#   740#
                   951#   979#   982#  1096#  1129#  1165#  1174#  1282#  1363#  1372#  1374#  1381#  1423#
                  1521#  1581#
$$CASE= 000000     281#
$$DST = 000000     281#
$$ELOC= 000402     281#   551#   556    561#   585#   589#   600#   610#   662#   670#   685#   690#   707#
                   714#   729#   740#   751#   753#   755#   757#   951#   971#   979#   982#   988    999#
                  1001   1012#  1096#  1117#  1129#  1150#  1165#  1172#  1174#  1178#  1282#  1300#  1363#
                  1372#  1381#  1388#  1390#  1394   1399#  1423#  1432   1437#  1521#  1539   1544#  1581#
                  1586#
$$ERFL= 000000     281#
$$FLAG= 000001     281#   534#   535#   537#   551#   553#   561#   585#   587#   589#   600#   602#   610#
                   651#   652#   654#   662#   664#   670#   680#   681#   683#   685#   687#   690#   707#
                   709#   714#   716#   729#   731#   740#   742#   751#   753#   755#   757#   951#   953#
                   971#   979#   981#   982#   984#   999#  1012#  1096#  1098#  1117#  1129#  1131#  1150#
                  1165#  1167#  1172#  1174#  1176#  1178#  1282#  1284#  1300#  1363#  1365#  1372#  1374#
                  1377#  1381#  1383#  1388#  1390#  1399#  1423#  1425#  1437#  1521#  1523#  1544#  1581#
                  1583#  1586#
```

```
$$FROM= 000000            281#    532#    567#    577#    649#    655#    659#    667#    672#    701#    705#    712#    727#
                          745#    759#    969#   1114#   1147#   1220#   1258#   1339#   1366#   1397#   1426#   1435#   1461#
                         1500#   1504#
$$LOC = 004444R           281#    536#    537     552#    553     586#    587     601#    602     653#    654     663#    664
                          682#    683     686#    687     708#    709     715#    716     730#    731     741#    742     952#
                          953     960#    981     983#    984    1015#   1016    1097#   1098    1122#   1123    1130#   1131
                         1166#   1167    1175#   1176    1283#   1284    1364#   1365    1375#   1376    1382#   1383    1424#
                         1425    1522#   1523    1582#   1583
$$LOCN= 000000            281#
$$REG = 177777            281#
$$RETU= 000000            281#
$$RTN1= 000000            281#
$$RTN2= 000000            281#
$$SRC = 000000            281#
$$TGSV= 000000            281#
$$TGS1= 000000            281#
$$TGS2= 000000            281#
$$TO  = 000000            281#    532#    533     567#    568     577#    578     649#    650     655#    656     659#    660
                          667#    668     672#    673     701#    702     705#    706     712#    713     727#    728     745#
                          746     759#    760     969#    970    1114#   1115    1147#   1148    1220#   1221    1258#   1259
                         1339#   1340    1366#   1367    1397#   1398    1426#   1427    1435#   1436    1461#   1462    1500#
                         1501    1504#   1585
$$$TAG= 050000            281#    679
.     = 005214R           518#    536     537     552     553     586     587     601     602     653     654     663     664
                          682     683     686     687     708     709     715     716     721     730     731     736     741
                          742     952     953     980     981     983     984    1015    1016    1097    1098    1122    1123
                         1130    1131    1166    1167    1175    1176    1283    1284    1364    1365    1375    1376    1382
                         1383    1424    1425    1522    1523    1582    1583    1618#

. ABS.  000000     000
        005214     001

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 XTTSAA0,XTSAA0/SOL/CRF:SYM=DDXCOM,SPMAC/ML,XTSAA0
 RUN-TIME:  34.26 1 SECONDS
 RUN-TIME RATIO: 230/63=3.6
 CORE USED:  16K  (31 PAGES)
```

```
XX      XX    TTTTTTTTTT      SSSSSSSS      AAAAAA        AAAAAA        000000
XX      XX    TTTTTTTTTT      SSSSSSSS      AAAAAA        AAAAAA        000000
XX      XX        TT          SS            AA    AA      AA    AA      00    00
XX      XX        TT          SS            AA    AA      AA    AA      00    00
  XX  XX          TT          SS            AA    AA      AA    AA      00  0000
  XX  XX          TT          SS            AA    AA      AA    AA      00  0000
    XX            TT            SSSSSS      AA    AA      AA    AA      00  00 00
    XX            TT            SSSSSS      AA    AA      AA    AA      00 00  00
  XX  XX          TT                  SS    AAAAAAAAAA    AAAAAAAAAA    0000   00
  XX  XX          TT                  SS    AAAAAAAAAA    AAAAAAAAAA    0000   00
XX      XX        TT                  SS    AA    AA      AA    AA      00     00
XX      XX        TT                  SS    AA    AA      AA    AA      00     00
XX      XX        TT          SSSSSSSS      AA    AA      AA    AA        000000
XX      XX        TT          SSSSSSSS      AA    AA      AA    AA        000000

  SSSSSSSS    EEEEEEEEEE      QQQQQQ
  SSSSSSSS    EEEEEEEEEE      QQQQQQ
SS            EE            QQ    QQ
SS            EE            QQ    QQ
SS            EE            QQ    QQ
SS            EE            QQ    QQ
  SSSSSS      EEEEEEEE      QQ    QQ
  SSSSSS      EEEEEEEE      QQ    QQ
        SS    EE            QQ  QQ QQ
        SS    EE            QQ  QQ QQ
        SS    EE            QQ    QQ
        SS    EE            QQ    QQ
SSSSSSSS    EEEEEEEEEE      QQQQ  QQ
SSSSSSSS    EEEEEEEEEE      QQQQ  QQ
```

```
*************************************************************************
 *  *  *  *  *    R E P O R T   P R O B L E M S ?   *  *  *  *  *  *  *
* If you have problems/questions with this report, please               *
*  A. Check the applicable items.                                       *
*     ___ Do not need this report, please discontinue                   *
*     ___ Interested in receiving report on microfiche                  *
*     ___ Interested in receiving report on 8 1/2 X 11 Xerox paper      *
*     ___ Address is wrong or incomplete (Enter correctly below)        *
*     ___ Have other problem/question; not sure whom to contact         *
*  B. Identify yourself and the report.          Phone:                 *
*     Name:                              Location (mail stop):          *
*                                                                       *
*     CC #:       Report name:                    Report #:             *
*                                                                       *
*  C. Mail entire page to: Corporate Information Services  PK3-2/F34    *
 *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
```

LPTSPL version 102(2263)/3(61) running on MTA064
 START User BEAUSOLEIL, [400,2043] Job XTSAA0 Seq. 5716 Date 31-Jan-79 11:28:03 Monitor IPC-F 603 [7C2] *START*
/TO:ML21-4:BEAUSO -- Distribution to ML21-4, slot 132
File: SAIL:XTSAA0.SEQ<057>[400,2043] Created: 31-Jan-79 10:40:00 Printed: 31-Jan-79 11:28:08
QUEUE Switches: /PRINT:ARROW /FILE:ASCII /COPIES:2 /SPACING:1 /LIMIT:450 /FORMS:UNROK