```
.REM    %
```

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DTTCA-B

PRODUCT NAME: TC11 DEVICE ROUTINE FOR MPG

DATE: APRIL 1976

MAINTAINER: SYSTEMS RELIABILITY

AUTHOR: C. E. HARPER


```
        %
```

```
                .SBTTL  REVISION HISTORY

;
;       APR 76  DTTCA-B RELEASE
;
;       DEC 75  WILL NOW DISPLAY THE UNIT # BYTE IN OCTAL FOR
;               INVALID UNIT # ERROR MESSAGES.
;
;       DEC 75  MADE CHANGES REQUIRED FOR THE MEMORY MANAGEMENT
;               VERSION OF MPG.
;
;       AUG 75  DTTCA-A INITIAIL RELEA'E
```

```
                                        .SBTTL  STANDARD DEVICE ROUTINE TABLE


                                        .TITLE  MAINDEC-11-DTTCA-B  TC11/TU56 DEVICE ROUTINE FOR MPG

                                        ;REVISION 'B'

                                        ;FILENAME OF "TTCAB0.MPG" ON MPG/XXDP MEDIA

                                        ;MACY11:  DTTCA' DTTCA'/CRF:SYM/DOC=DTTCA'.P11
                                        ;LNKX11:  DTTCA'.MPG/B:0+DTTCA'/E
                                        ;PAPER TAPE:  PUNCH DTTCA'.MPG/FILE:ELEV

          000000'                       .CSECT  TC11
                                        .DSABL  GBL


                                        ;THE FOLLOWING TABLE IS IN THE STANDARDIZED FORMAT REQUIRED
                                        ;TO INTERFACE WITH MPG.

000000' 005276          LOCZ:   .WORD   DVREND-.                ;DEVICE ROUT SIZE IN BYTES
000002' 000000          DFLGWD: .WORD   0                       ;DEVICE ROUT FLAGWORD
                                                                ;   BIT 15 = "NOWAIT" FLAG
                                                                ;   BIT 11 = 0 - FWD, 1 - REV
                                                                ;   BIT  3 = BLK SRCH ERROR
                                                                ;   BIT  1 = DO I/O TERMINATION
                                                                ;   BIT  0 = ERROR ON I/O CMND
000004' 000000          BLK:    .WORD   0                       ;CURRENT STARTING BLOCK NUMBER
000006' 000000                  .WORD   0                       ;INTERFACE WORD # 2 (NOT USED)
000010' 000000                  .WORD   0                       ;INTERFACE WORD # 3 (NOT USED)
000012' 000000                  .WORD   0                       ;INTERFACE WORD # 4 (NOT USED)
000014' 000000                  .WORD   0                       ;INTERFACE WORD # 5 (NOT USED)
000016' 000000                  .WORD   0                       ;INTERFACE WORD # 6 (NOT USED)
000020' 000001          SIZE:   .WORD   1                       ;# OF BYTES TRANSFERRED / UNIMAP FLG
000022' 000000          ERR:    .WORD   0                       ;ERROR ON LAST I/O INDICATOR
000024' 177340          DREGAD: .WORD   177340                  ;FIRST DEVICE REGISTER ADR
000026' 000214          IVCTAD: .WORD   214                     ;INTERRUPT VECTOR ADR
000030' 000300          PSWD:   .WORD   300                     ;INT PROC STATUS WORD (BR 6)
000032' 000000                  .WORD   0                       ;NOT USED
000034' 000710                  .WORD   HSKEEP-.                ;HOUSEKEEPING ROUT REL ADR
000036' 000742                  .WORD   REPORT-.                ;REPORT ROUT REL ADR
000040' 001360                  .WORD   KILL-.                  ;KILL ROUT REL ADR
000042' 000654                  .WORD   DATAER-.                ;DATA ERROR COUNTER REL ADR
000044' 001254                  .WORD   TOUTER-.                ;TIME OUT ERROR ROUT REL ADR
000046' 000000          CIOBSY. .WORD   0                       ;I/O BUSY BRANCH ADR
000050' 000000          CUPGER: .WORD   0                       ;DEVICE ERROR BRANCH ADR
000052' 000000          ULIST:  .WORD   0                       ;USER MODE PRINT ROUTINE BRANCH ADR
000054' 000000          CLIST:  .WORD   0                       ;CMND MODE PRINT ROUTINE BRANCH ADR
000056' 000000          BINASC: .WORD   0                       ;CONVERT BINARY TO ASCII ROUT BR ADR
000060' 000000          BTASLZ: .WORD   0                       ;CONVERT BINARY TO DECIMAL ASCII BR ADR
000062' 000000          DECASC: .WORD   0                       ;CONVERT PACKED DECIMAL TO ASCII BR ADR
000064' 000000          CSYSFW: .WORD   0                       ;MPG SYSTEM FLAGWORD ADR
000066' 000000          SETVEC: .WORD   0                       ;SET INT VECT ROUT BR ADR
000070' 000000          CLRVEC: .WORD   0                       ;CLEAR INT VECTOR ROUT BR ADR
000072' 000000          TSTVEC: .WORD   0                       ;TEST INT VECTOR ROUT BR ADR
```

```
                                                RTNINT: .WORD   0                       ;RETURN FROM INT ROUT BR ADR
                                                GETBYT: .WORD   0                       ;GET DATA BYTE ROUT BR ADR
                                                PUTBYT: .WORD   0                       ;PUT DATA BYTE ROUT BR ADR
                                                        .WORD   DVREGS-.                ;ADR OF DEVICE REGISTER NAMES
                                                        .WORD   DVCMDS-.                ;ADR OF DEVICE FUNCTIONS
                                                        .WORD   DVPKTE-.                ;ADR OF PACK TBL EXTENSION
                                                        .WORD   DVMVTE-.                ;ADR OF MODEL VECTOR TBL EXTEN.
                                                        .WORD   DVCPTE-.                ;ADR OF COMPILER TBL EXTEN.
                                                        .WORD   DVIWST-.                ;ADR OF DEV INTERFACE WD SYM TBL


                                                DVREGS: .ASCII  /TCST/                  ;VALID DEVICE REGISTER NAMES &
                                                        .WORD   0                       ;THEIR POSITIONS RELATIVE TO
                                                        .ASCII  /TCCM/                  ;THE DEVICE REGISTERS BASE ADDRESS.
                                                        .WORD   2
                                                        .ASCII  /TCWC/
                                                        .WORD   4
                                                        .ASCII  /TCBA/
                                                        .WORD   6
                                                        .ASCII  /TCDT/
                                                        .WORD   10
                                                DVREGE= .


                                                DVCMDS: .BYTE   120,201                 ;VALID DEVICE FUNCTIONS
                                                        .WORD   READ-.                  ;  FLAG BYTE:
                                                        .BYTE   130,201                 ;     BIT 7 = NPR DEV
                                                        .WORD   WRITE-.                 ;     BIT 3 = MASSBUS DEV
                                                        .BYTE   376,0                   ;     BIT 0 = 2 WORDS FOR ADR
                                                        .WORD   NOWAIT-.                ;             (18 BIT ADRS.
                                                        .BYTE   375,0
                                                        .WORD   WAIT-.
                                                        .BYTE   374,0
                                                        .WORD   REPORT-.
                                                        .BYTE   373,0
                                                        .WORD   REPORT-.
                                                        .BYTE   372,0
                                                        .WORD   FWD-.
                                                        .BYTE   371,0
                                                        .WORD   REV-.
                                                        .BYTE   370,0
                                                        .WORD   RDNUM-.
                                                        .BYTE   367,0
                                                        .WORD   RDALL-.
                                                        .BYTE   366,0
                                                        .WORD   WRALL-.
                                                        .BYTE   365,0
                                                        .WORD   WRTM-.
                                                        .BYTE   364,0
                                                        .WORD   STOP-.
                                                        .BYTE   363,0
                                                        .WORD   STPALL-.
                                                        .WORD   177777                  ;TABLE TERMINATOR


                                                DVPKTE: .ASCII  /NOWAIT/                ;PACK TABLE EXTENSION
                                                        .BYTE   376,0
                                                        .ASCII  /  WAIT/
```

```
000264'    375     000                      .BYTE   375,0
000266'  052123  052101  051525             .ASCII  /STATUS/
000274'    374     000                      .BYTE   374,0
000276'  047503  047125  051524             .ASCII  /COUNTS/
000304'    373     000                      .BYTE   373,0
000306'  020040  043040  042127             .ASCII  /   FWD/
000314'    372     000                      .BYTE   372,0
000316'  020040  051040  053105             .ASCII  /   REV/
000324'    371     000                      .BYTE   371,0
000326'  051040  047104  046525             .ASCII  / RDNUM/
000334'    370     000                      .BYTE   370,0
000336'  051040  040504  046114             .ASCII  / RDALL/
000344'    367     000                      .BYTE   367,0
000346'  053440  040522  046114             .ASCII  / WRALL/
000354'    366     000                      .BYTE   366,0
000356'  020040  051127  046524             .ASCII  /   WRTM/
000364'    365     000                      .BYTE   365,0
000366'  020040  052123  050117             .ASCII  /   STOP/
000374'    364     000                      .BYTE   364,0
000376'  052123  040520  046114             .ASCII  /STPALL/
000404'    363     000                      .BYTE   363,0

000406'  000376  000632          DVMVTE:    .WORD   376,LNWAIT-LOCZ          ;MODEL VECTOR TABLE EXTEN.
000412'  000375  000632                     .WORD   375,LWAIT-LOCZ
000416'  000374  000632                     .WORD   374,LSTATS-LOCZ
000422'  000373  000632                     .WORD   373,LCOUNT-LOCZ
000426'  000372  000632                     .WORD   372,LFWD-LOCZ
000432'  000371  000632                     .WORD   371,LREV-LOCZ
000436'  000370  000633                     .WORD   370,LRDNUM-LOCZ
000442'  000367  000633                     .WORD   367,LRDALL-LOCZ
000446'  000366  000642                     .WORD   366,LWRALL-LOCZ
000452'  000365  000642                     .WORD   365,LWRTM-LOCZ
000456'  000364  000632                     .WORD   364,LSTOP-LOCZ
000462'  000363  000632                     .WORD   363,LSTALL-LOCZ


                                 ;
                                 ;          COMPILER TABLE EXTENSION
                                 ;

000466'    003     376           DVCPTE:    .BYTE   3,376                    ;NO WAIT
000470'  004537  000012                     .WORD   4537,10.
000474'    003     375                      .BYTE   3,375                    ;WAIT
000476'  004537  000012                     .WORD   4537,10.
000502'    004     374                      .BYTE   4,374                    ;STATUS
000504'  004537  000012  001002             .WORD   4537,10.,1002
000512'    004     373                      .BYTE   4,373                    ;COUNTS
000514'  004537  000012  001001             .WORD   4537,10.,1001
000522'    003     372                      .BYTE   3,372                    ;FORWARD
000524'  004537  000012                     .WORD   4537,10.
000530'    003     371                      .BYTE   3,371                    ;REVERSE
000532'  004537  000012                     .WORD   4537,10.
000536'    005     370                      .BYTE   5,370                    ;READ NUMBER
000540'  004537  000012  000000             .WORD   4537,10.,0,2
000546'  000002
000550'    005     367                      .BYTE   5,367                    ;READ ALL
000552'  004537  000012  000000             .WORD   4537,10.,0,2
000560'  000002
```

```
000562'    005      366                       .BYTE   5,366              ;WRITE ALL
000564'  004537   000012   000000             .WORD   4537,10.,0,2
000572'  000002
000574'    005      365                       .BYTE   5,365              ;WRITE TIMING & MARK
000576'  004537   000012   000000             .WORD   4537,10.,0,2
000604'  000002
000606'    003      364                       .BYTE   3,364              ;STOP
000610'  004537   000012                      .WORD   4537,10.
000614'    003      363                       .BYTE   3,363              ;STOP ALL
000616'  004537   000012                      .WORD   4537,10.

                                      ;
                                      ;       DEVICE INTERFACE WORD SYMBOL TABLE
                                      ;
000622'  046102   020113             DVIWST:  .ASCII  /BLK /
000626'  000004                               .WORD   DEVIW1
000630'  177777                               .WORD   177777             ;END OF TABLE

                                      ;
                                      ;       MODEL STATEMENT TABLE
                                      ;
000632'                              LNWAIT:
000632'                              LWAIT:
000632'                              LSTATS:
000632'                              LCOUNT:
000632'                              LFWD:
000632'                              LREV:
000632'                              LSTOP:
000632'    000                       LSTALL:  .BYTE   0
000633'                              LRDNUM:
000633'    377   047111   047524     LRDALL:  .ASCIZ  <377>/INTO/<377>
000640'  000377
000642'                              LWRALL:
000642'  043377   047522   177515    LWRTM:   .ASCIZ  <377>/FROM/<377>
000650'    000
         000652'                              .EVEN

         000652'                     HSKPST=  .
         000652'                     ISTAT=   .                          ;STORAGE FOR DEV REG'S AT INT
000652'  000000                               .WORD   0
000654'  000000                               .WORD   0
000656'  000000                               .WORD   0
000660'  000000                               .WORD   0
000662'  000000                               .WORD   0

000664'  000005                      CSTAT:   .BLKW   5                  ;DEV REG CURRENT VALUES STORAGE

000676'  000000                      BYRD:    .WORD   0                  ;BYTES READ COUNT (READ, RDALL, & RDNUM)
000700'  000000                               .WORD   0
000702'  000000                      BYWR:    .WORD   0                  ;BYTES WRITTEN COUNT (WRITE, WRALL, &
000704'  000000                               .WORD   0                  ;  WRTM)
000706'  000000                      RDCNT:   .WORD   0                  ;READ CMND COUNT (READ, RDALL, & RDNUM)
000710'  000000                      WRCNT:   .WORD   0                  ;WRITE CMND COUNT (WRITE, WRALL, & WRTM)
000712'  000000                      MISCNT:  .WORD   0                  ;MISC. CMND COUNT (STOP & STPALL)
000714'  000000                      ERRCNT:  .WORD   0                  ;DEVICE ERRORS COUNT
000716'  000000                      DATAER:  .WORD   0                  ;DATA ERRORS COUNT
```

```
000720' 000000                  INTCNT: .WORD   0                       ;INTERRUPTS COUNT

000722' 000000                  TOECNT: .WORD   0                       ;# OF ENTRIES INTO T/O ERROR ROUT
000724' 000000                  ERRADR: .WORD   0                       ;CURR ADR IN USER PROG
000726' 000000                  CNTADR: .WORD   0                       ;ADR OF BYTE COUNT TOTALS
000730' 000000                  CURFLG: .WORD   0                       ;FLAG WORD OF CURR CMND
000732' 000000                  CURCNT: .WORD   0                       ;WORD CNT FOR CURR CMND
000734' 000000                  CURCMD: .WORD   0                       ;CURRENT BLK ORIENTED CMND
000736' 000000                  INBLKN: .WORD   0                       ;INITIAL BLK # FOR SEARCH
000740' 000000                  FINCNT: .WORD   0                       ;FINAL WORD COUNT (TCWC)
000742' 000000                  REVCNT: .WORD   0                       ;TAPE DIRECTION REVERSAL CNT
        000744'                 HSKPEN= .

        000000                  XXXX=   0                               ;VALUE TO BE TAILORED BY DEV ROUT
```

```
                                        .SBTTL  TC11 SUPPORT ROUTINES ENTERED FROM MPG


                                        ;DEVICE ROUTINE HOUSEKEEPING

                                                ;JSR    R5,HSKEEP       S/R CALL
                                                ;.WORD  0 OR 1          0 = DO HSKP PER OPSW
                                                ;                       1 = UNCOND. DO HSKP
                                                ;R2 = PROG'S OPSW
                                                ;DESTROYS R0,R1

000744' 005725          HSKEEP: TST     (R5)+                   ;UNCONDITIONALLY DO HSKP?
000746' 001003                  BNE     10$                     ;N,Y-10$
000750' 032702  000004          BIT     #HSKPEP,R2              ;OPSW SPECIFY EACH PASS HSKP?
000754' 001010                  BNE     30$                     ;Y,N-30$
000756' 010700          10$:    MOV     PC,R0                   ;SET UP FIRST WC ADR
000760' 062700  177672          ADD     #HSKPST-.,R0
000764' 012701  000035          MOV     #HSKPEN-HSKPST/2,R1     ;SET UP # OF WORDS
000770' 005020          20$:    CLR     (R0)+                   ;HSKP ALL NECESSARY AREAS
000772' 005301                  DEC     R1
000774' 001375                  BNE     20$
000776' 000205          30$:    RTS     R5                      ;EXIT IN-LINE


                                        ;TC11 REPORT ROUTINE

                                                ;JSR    R5,REPORT       S/R CALL
                                                ;.WORD  FLGWD           FLAGWORD
                                                ;                         BIT 15 = CMND MODE CALL
                                                ;                         BIT  9 = PROG STMNT CALL
                                                ;                         BIT  1 = DO STATUS REPORT
                                                ;                         BIT  0 = DO COUNTS REPORT

001000' 004067  003120  REPORT: JSR     R0,SAVREG               ;SAVE REG'S R0 - R5
001004' 032715  177776          BIT     #177776,(R5)            ;DISPLAYING CNTS AT END OF
001010' 001012                  BNE     10$                     ;PROG PASS? (Y,N-10$)
001012' 010700                  MOV     PC,R0                   ;SET UP ADR OF CNTS
001014' 062700  177662          ADD     #BYRD-.,R0
001020' 012701  000012          MOV     #10.,R1                 ;GET # OF CNT WORDS
001024' 005720          5$:     TST     (R0)+                   ;THIS CNT WORD = 0?
001026' 001003                  BNE     10$                     ;Y,N-10$
001030' 005301                  DEC     R1                      ;DECR WORD CNT
001032' 001374                  BNE     5$                      ;CK'ED ALL WORDS? (Y,N-5$)
001034' 000513                  BR      DVREX                   ;GO TO EXIT -- ALL CNTS ARE 0'S
001036' 004767  003114  10$:    JSR     PC,SUPTAD               ;SET UP PROG TBL ADR IN R3
001042' 012504                  MOV     (R5)+,R4                ;GET FLAGWORD
001044' 032704  000002          BIT     #2,R4                   ;GOING TO DO STATUS DISPLAY?
001050' 001443                  BEQ     DISCNT                  ;Y,N-DISCNT
001052' 004567  003124          JSR     R5,STSTAT               ;GO STORE STATUS REG'S
001056' 177606                  .WORD   CSTAT-.
001060' 010700                  MOV     PC,R0                   ;SET UP ADR OF REG'S AT
001062' 062700  177570          ADD     #ISTAT-.,R0             ;LAST INT
001066' 012701  000005          MOV     #5,R1                   ;SET UP # OF REG'S
001072' 005720          20$:    TST     (R0)+                   ;ALL REG'S = 0?
001074' 001003                  BNE     30$                     ;N,Y-40$
001076' 005301                  DEC     R1
```

```
345  001100' 001374                      BNE     20$
346  001102' 000412                      BR      40$
347  001104' 004767 003116       30$:    JSR     PC,DISUNM               ;DISPLAY CURR UNIT #
348  001110' 004567 003316               JSR     R5,PRINT                ;ISSUE 'AT LAST INT' MSG
349  001114' 003437                      .WORD   ATIMSG-.
350  001116' 000031                      .WORD   25.
351  001120' 004567 003172               JSR     R5,DISPST               ;GO DISPLAY STATUS AT LAST INT
352  001124' 177526                      .WORD   ISTAT-.
353  001126' 000402                      BR      45$                     ;CONTINUE DISPLAY
354  001130' 004767 003072       40$:    JSR     PC,DISUNM               ;DISPLAY CURR UNIT #
355  001134' 004567 003272       45$:    JSR     R5,PRINT                ;ISSUE 'CURRENTLY' MSG
356  001140' 003444                      .WORD   CURMSG-.
357  001142' 000012                      .WORD   10.
358  001144' 004567 003146               JSR     R5,DISPST               ;GO DISPLAY CURRENT STATUS
359  001150' 177514                      .WORD   CSTAT-.
360  001152' 004767 003230               JSR     PC,PRTIWD               ;GO DISPLAY INFO WORDS
361  001156' 000402                      BR      DISCT1                  ;CHECK FOR COUNTS DISPLAY
362  001160' 004767 003042       DISCNT: JSR     PC,DISUNM               ;DISPLAY CURR UNIT #
363  001164' 032704 000001       DISCT1: BIT     #1,R4                   ;DISPLAY COUNTS?
364  001170' 001431                      BEQ     RPTEND                  ;Y,N-RPTEND
365  001172' 012700 000012               MOV     #10.,R0                 ;SET UP # OF WORDS
366  001176' 010701                      MOV     PC,R1                   ;SET UP ADR OF CNTS
367  001200' 062701 177476               ADD     #BYRO-.,R1
368  001204' 010702                      MOV     PC,R2                   ;SET UP TBL ADR
369  001206' 062702 000066               ADD     #REPTBL-.,R2
370  001212' 012267 000012       RPTLP:  MOV     (R2)+,RPTBAS            ;MOV MSG ADR TO S/R LINKAGE
371  001216' 004067 002702               JSR     R0,SAVREG               ;SAVE ALL REG'S
372  001222' 011100                      MOV     (R1),R0                 ;GET CURRENT COUNT
373  001224' 004577 176626               JSR     R5,@BINASC              ;CONVERT IT TO ASCII
374  001230' 000000       RPTBAS: .WORD   XXXX
375  001232' 004067 002702               JSR     R0,RESREG               ;RESTORE REG'S
376  001236' 005721                      TST     (R1)+                   ;POINT AT NXT CNT
377  001240' 005300                      DEC     R0                      ;DONE ALL WORDS?
378  001242' 001363                      BNE     RPTLP                   ;Y,N-RPTLP
379  001244' 004567 003162               JSR     R5,PRINT                ;GO ISSUE COUNTS MSG
380  001250' 003434                      .WORD   CNTSMG-.
381  001252' 000221                      .WORD   CNTSEN-CNTSMG
382  001254' 004567 003152       RPTEND: JSR     R5,PRINT                ;ISSUE "END OF REPORT" MSG
383  001260' 003336                      .WORD   RENDMG-.
384  001262' 177763                      .WORD   -13.
385  001264' 004067 002650       DVREX:  JSR     R0,RESREG               ;RESTORE REGISTERS
386  001270' 005725                      TST     (R5)+                   ;SET UP RETURN POINT
387  001272' 000205                      RTS     R5                      ;EXIT IN-LINE
388
389
390  001274' 003470       REPTBL: .WORD   BCMRO-RPTBAS
391  001276' 003476               .WORD   BCMRO+6-RPTBAS
392  001300' 003512               .WORD   BCMWR-RPTBAS
393  001302' 003520               .WORD   BCMWR+6-RPTBAS
394  001304' 003545               .WORD   CMDCRO-RPTBAS
395  001306' 003560               .WORD   CMDCWR-RPTBAS
396  001310' 003575               .WORD   CMDCMS-RPTBAS
397  001312' 003624               .WORD   CNTERR-RPTBAS
398  001314' 003641               .WORD   CNTDER-RPTBAS
399  001316' 003667               .WORD   CNTINT-RPTBAS
400
```

```
                                        ;TIMEOUT ERROR ROUTINE
                                        ;
                                        ;       JSR     R5,TOUTER       S/R CALL
                                        ;
001320' 005267  177376          TOUTER: INC     TOECNT                  ;INCR # OF TIMEOUTS THAT OCCURRED
001324' 026727  177372  000007          CMP     TOECNT,#7               ;THIS SEVENTH TIMEOUT IN A ROW?
001332' 001031                          BNE     TOUTEX                  ;Y,N-TOUTEX
001334' 004067  002564                  JSR     R0,SAVREG               ;SAVE ALL REGISTERS
001340' 004767  002612                  JSR     PC,SUPTAD               ;SET UP TCCM & PROG TBL ADR'S
001344' 004567  002632                  JSR     R5,STSTAT               ;STORE CURRENT STATUS
001350' 177314                          .WORD   CSTAT-.
001352' 004567  002516                  JSR     R5,TVECT                ;CK IF I HAVE VECTOR CONTROL
001356' 000404                          BR      10$                     ;BR IF I DON'T
001360' 112714  000011                  MOVB    #11,(R4)                ;RESET INT ENABLE & STOP THE TAPE
001364' 004767  002460                  JSR     PC,RINTV                ;RESET THE INTERRUPT VECTOR
001370' 042713  000010          10$:    BIC     #WT4IOT,(R3)            ;RESET WAITING FOR I/O FLAG
001374' 004567  001674                  JSR     R5,ERRC$1               ;ISSUE TIMEOUT ERROR MSG
001400' 001553                          .WORD   IOTO-ERMBAS
001402' 000016                          .WORD   14.
001404' 004067  002530                  JSR     R0,RESREG               ;RESTORE REGISTERS
001410' 012605                          MOV     (SP)+,R5                ;REMOVE RETURN ADR
001412' 000177  176432                  JMP     @CUPGER                 ;GO TO ERROR EXIT
001416' 000205                  TOUTEX: RTS     R5                      ;EXIT IN-LINE
                                        ;
                                        ;KILL USER PROGRAM ROUTINE
                                        ;
                                        ;       JSR     R5,KILL         S/R CALL
                                        ;R3 MUST CONTAIN PROG TBL ADR
                                        ;DESTROYS R0,R1
                                        ;
001420' 004567  002450          KILL:   JSR     R5,TVECT                ;CK IF I HAVE VECTOR CONTROL
001424' 000407                          BR      KILLEX                  ;BR IF I DON'T
001426' 016701  176372                  MOV     DREGAD,R1               ;GET DEV REG ADR
001432' 112761  000011  000002          MOVB    #11,2(R1)               ;RESET INT ENABLE & STOP THE TAPE
001440' 004767  002404                  JSR     PC,RINTV                ;RESET INT VECTOR INFO
001444' 000205                  KILLEX: RTS     R5                      ;EXIT IN-LINE
```

```
                                        .SBTTL  TC11 FUNCTION ROUTINES


                                        ;"WAIT" FUNCTION ROUTINE

                                                ;JSR    R5,WAIT         FUNCTION CALL

001446' 042767  100000  176326  WAIT:   BIC     #100000,DFLGWD          ;RESET THE "NOWAIT" FLAG
001454' 004767  001506                  JSR     PC,CKDBSY               ;WAIT IF BUSY & DO TERMINATION
001460' 004767  002364                  JSR     PC,RINTV                ;RESET THE INTERRUPT VECTOR
001464' 000205                          RTS     R5                      ;EXIT IN-LINE


                                        ;"NOWAIT" FUNCTION ROUTINE

                                                ;JSR    R5,NOWAIT       FUNCTION CALL

001466' 052767  100000  176306  NOWAIT: BIS     #100000,DFLGWD          ;SET THE "NOWAIT" FLAG
001474' 000205                  FUNCEX: RTS     R5                      ;EXIT IN-LINE


                                        ;"FWD" FUNCTION ROUTINE

                                                ;JSR    R5,FWD          FUNCTION CALL

001476' 042767  004000  176276  FWD:    BIC     #4000,DFLGWD            ;RESET THE REVERSE FLAG
001504' 000205                          RTS     R5                      ;EXIT IN-LINE


                                        ;"REV" FUNCTION ROUTINE

                                                ;JSR    R5,REV          FUNCTION CALL

001506' 052767  004000  176266  REV:    BIS     #4000,DFLGWD            ;SET THE REVERSE FLAG
001514' 000205                          RTS     R5                      ;EXIT IN-LINE


                                        ;"READ" FUNCTION ROUTINE

                                                ;JSR    R5,READ         FUNCTION CALL
                                                ;.WORD  ADR             DATA ADDRESS (BITS 16 & 17)
                                                ;.WORD  ADR             DATA ADDRESS (BITS 0 - 15)
                                                ;.WORD  CNT             BYTE COUNT
                                                ;.WORD  DEV             (NOT USED)

001516' 012701  000105  READ:   MOV     #105,R1                 ;SET UP READ CMND CODE
001522' 012702  000011          MOV     #011,R2                 ;SET UP READ FLAG WORD
001526' 004767  001434  RDCOM:  JSR     PC,CKDBSY               ;GO CK IF DEV IS BUSY
001532' 005267  177150          INC     RDCNT                   ;ADD 1 TO READ CMND CNT
001536' 010700                  MOV     PC,R0                   ;SET UP ADR OF BYTES READ CNT
001540' 062700  177140          ADD     #BYRD+2-.,R0
001544' 000456                  BR      CMDCOM                  ;GO TO CMND COMMON PROCESSING
```

```
                                        ;"WRITE" FUNCTION ROUTINE

                                                ;JSR    R5,WRITE        FUNCTION CALL
                                                ;.WORD  ADR             DATA ADDRESS (BITS 16 & 17)
                                                ;.WORD  ADR             DATA ADDRESS (BITS 0 - 15)
                                                ;.WORD  CNT             BYTE COUNT
                                                ;.WORD  DEV             (NOT USED)

001546' 012701  000115          WRITE:  MOV     #115,R1                 ;SET UP WRITE CMND CODE
001552' 012702  000011                  MOV     #011,R2                 ;SET UP CMND FLAG WORD
001556' 004767  001404          WRCOM:  JSR     PC,CKDBSY               ;GO CK IF DEV IS BUSY
001562' 005267  177122                  INC     WRCNT                   ;ADD 1 TO WRITE CMND CNT
001566' 010700                          MOV     PC,R0                   ;SET UP ADR OF BYTES WRITTEN CNT
001570' 062700  177114                  ADD     #BYWR+2-.,R0
001574' 000442                          BR      CMDCOM                  ;GO TO CMND COMMON PROCESSING


                                        ;"RDNUM" FUNCTION ROUTINE

                                                ;JSR    R5,RDNUM        FUNCTION CALL
                                                ;.WORD  ADR             DATA ADDRESS
                                                ;.WORD  CNT             BYTE COUNT

001576' 012701  000103          RDNUM:  MOV     #103,R1                 ;SET UP RDNUM CMND CODE
001602' 012702  000102                  MOV     #102,R2                 ;SET UP CMND FLAG WORD
001606' 000747                          BR      RDCOM                   ;GO TO COMMON READ PROCESSING


                                        ;"RDALL" FUNCTION ROUTINE

                                                ;JSR    R5,RDALL        FUNCTION CALL
                                                ;.WORD  ADR             DATA ADDRESS
                                                ;.WORD  CNT             BYTE COUNT

001610' 012701  000007          RDALL:  MOV     #007,R1                 ;SET UP RDALL CMND CODE
001614' 012702  000232                  MOV     #232,R2                 ;SET UP CMND FLAG WORD
001620' 000742                          BR      RDCOM                   ;GO TO COMMON READ PROCESSING


                                        ;"WRALL" FUNCTION ROUTINE

                                                ;JSR    R5,WRALL        FUNCTION CALL
                                                ;.WORD  ADR             DATA ADDRESS
                                                ;.WORD  CNT             BYTE COUNT

001622' 012701  000017          WRALL:  MOV     #017,R1                 ;SET UP WRALL CMND CODE
001626' 012702  000032                  MOV     #032,R2                 ;SET UP CMND FLAG WORD
001632' 000751                          BR      WRCOM                   ;GO TO COMMON WRITE PROCESSING
```

```
                                        ;"WRTM" FUNCTION ROUTINE
                                        ;
                                        ;JSR     R5,WRTM         FUNCTION CALL
                                        ;.WORD   ADR             DATA ADDRESS
                                        ;.WORD   CNT             BYTE COUNT
                                        ;
001634' 012701  000013          WRTM:   MOV     #013,R1         ;SET UP WRTM CMND CODE
001640' 012702  000022                  MOV     #022,R2         ;SET UP CMND FLAG WORD
001644' 000744                          BR      WRCOM           ;GO TO COMMON WRITE PROGESSING
                                        ;
                                        ;"STOP" FUNCTION ROUTINE
                                        ;
                                        ;JSR     R5,STOP         FUNCTION CALL
                                        ;
001646' 012701  000111          STOP:   MOV     #111,R1         ;SET UP STOP CMND CODE
001652' 012702  000044                  MOV     #044,R2         ;SET UP CMND FLAG WORD
001656' 004767  001304          MISCOM: JSR     PC,CKDBSY       ;GO CK IF DEV IS BUSY
001662' 005267  177024                  INC     MISCNT          ;ADD 1 TO MISC. CMND CNT
001666' 000405                          BR      CMDCOM          ;GO TO CMND COMMON PROCESSING
                                        ;
                                        ;"STPALL" FUNCTION ROUTINE
                                        ;
                                        ;JSR     R5,STPALL       FUNCTION CALL
                                        ;
001670' 012701  000101          STPALL: MOV     #101,R1         ;SET UP STPALL CMND CODE
001674' 012702  000044                  MOV     #044,R2         ;SET UP CMND FLAG WORD
001700' 000766                          BR      MISCOM          ;GO TO MISC. CMND COM PROCESSING
```

```
                                        ;COMMAND COMMON PROCESSING ROUTINE
                                        ;R4 = ADR OF TCCM DEV REG
                                        ;R3 = PROG TBL ADR
                                        ;R2 = COMMAND FLAG WORD
                                        ;R1 = COMMAND CODE
                                        ;R0 = ADR OF BYTE COUNT TOTALS, IF APPLICABLE

                                        ;CMND FLAGWORD FORMAT:

                                        ;BIT  7 = "RDALL" CMND
                                        ;BIT  6 = "RDNUM" CMND
                                        ;BIT  5 = DON'T ISSUE STOP AT FINAL INT
                                        ;BIT  4 = NON-INTERRUPT DATA SERVICE
                                        ;BIT  3 = BLOCK SEARCH BEFORE CMND
                                        ;BIT  2 = DO NOT INCREMENT BYTE COUNTS
                                        ;BIT  1 = 2 ARGUMENT CMND
                                        ;BIT  0 = 4 ARGUMENT CMND

001702' 010067 177020          CMDCOM: MOV     R0,CNTADR       ;SAVE ADR OF BYTE COUNT
001706' 010267 177016                  MOV     R2,CURFLG       ;SAVE FLAGWD FOR TERMINATION
001712' 005067 176104                  CLR     ERR             ;RESET THE ERROR INDICATOR
001716' 032702 000003                  BIT     #3,R2           ;THIS CMND HAVE ARGUMENTS?
001722' 001430                         BEQ     10$             ;Y,N-10$
001724' 062704 000004                  ADD     #4,R4           ;POINT AT BUS ADR REG
001730' 032702 000001                  BIT     #1,R2           ;4 ARGUMENT CMND?
001734' 001406                         BEQ     5$              ;Y,N-5$
001736' 012500                         MOV     (R5)+,R0        ;GET BITS 16 & 17 OF BUS ADR
001740' 006300                         ASL     R0              ;ALIGN THEM TO CORRECT
001742' 006300                         ASL     R0              ;BIT POSITIONS
001744' 006300                         ASL     R0
001746' 006300                         ASL     R0
001750' 050001                         BIS     R0,R1           ;SET THEM INTO CMND CODE WORD
001752' 012514                 5$:     MOV     (R5)+,(R4)      ;GET BUS ADR BITS 0 - 15
001754' 012544                         MOV     (R5)+,-(R4)     ;GET BYTE COUNT
001756' 006214                         ASR     (R4)            ;MAKE IT A WORD COUNT
001760' 011467 176746                  MOV     (R4),CURCNT     ;SAVE WORD COUNT
001764' 011467 176750                  MOV     (R4),FINCNT     ;INITIALIZE FINAL CNT TO SAME
001770' 005414                         NEG     (R4)            ;MAKE IT NEGATIVE
001772' 005744                         TST     -(R4)           ;REALIGN REG ADR TO TCCM
001774' 032702 000001                  BIT     #1,R2           ;4 ARGUMENT CMND?
002000' 001401                         BEQ     10$             ;Y,N-10$
002002' 005725                         TST     (R5)+           ;BYPASS FOURTH ARGUMENT
002004' 010167 176724          10$:    MOV     R1,CURCMD       ;SAVE CURR CMND CODE
002010' 116300 000035                  MOVB    PCURDV(R3),R0   ;GET CURR DEV #
002014' 020027 000007                  CMP     R0,#7           ;INV DEV #?
002020' 101405                         BLOS    20$             ;Y,N-20$
002022' 004567 001240                  JSR     R5,ERRCS        ;GO REPORT INV DEV # ERROR
002026' 001701                         .WORD   INVDVN-ERRBAS
002030' 000012                         .WORD   10.
002032' 000427                         BR      30$             ;GO TO ERR RETN
002034' 110064 000001          20$:    MOVB    R0,1(R4)        ;PUT DEV # IN TCCM BITS 8 THRU 10
002040' 016700 175736                  MOV     DFLGWD,R0       ;GET DEV ROUT FLGWD
002044' 042700 173777                  BIC     #173777,R0      ;RESET ALL BITS EXCEPT REV FLAG
002050' 050014                         BIS     R0,(R4)         ;SET UP TAPE DIRECTION
```

```
002052' 032702 000010          BIT   #10,R2          ;THIS A BLK SEARCH TYPE OF CMND?
002056' 001456                 BEQ   SOS             ;Y,N-SOS
002060' 016767 175720 176650   MOV   BLK,INBLKN      ;INITIALIZE BLK # FOR SEARCH S/R
002066' 012701 000103          MOV   #103,R1         ;SET UP "RDMUM" CMND CODE
002072' 026727 175706 001101   CMP   BLK,#577.       ;IS BLOCK # VALID?
002_30' 101415                 BLOS  SOS             ;N,Y-SOS
002102' 004567 001160          JSR   R5,ERRCS        ;REPORT INV BLK # ERROR
002106' 001713                 .WORD INVBKN-ERMBAS
002110' 000011                 .WORD 9.
002112' 005267 176600    30S:  INC   DATAER          ;ADD 1 TO DATA ERR CNT
002116' 012767 000001 175676   MOV   #1,ERR          ;SET THE ERROR INDICATOR
002124' 005367 176564          DEC   ERRCNT          ;REMOVE THE 1 ADDED TO DEV ERR CNT
002130' 000177 175714          JMP   @CUPGER         ;GO TO MPG ERR RETN POINT
002134' 042767 000011 175640 50S:  BIC   #11,DFLGWD  ;RESET THE ERROR FLAGS
002142' 005067 176574          CLR   REVCNT          ;RESET TAPE REVERSAL CNT
002146' 032702 000020          BIT   #20,R2          ;NON-INTERRUPT SERVICING?
002152' 001025                 BNE   NONINT          ;N,Y-NONINT
002154' 005063 000030          CLR   PTOCNT(R3)      ;INITIALIZE TIMEOUT COUNTER
002160' 005067 176536          CLR   TOECNT          ;RESET # OF TIMEOUTS
002164' 052767 000002 175610   BIS   #2,DFLGWD       ;SET THE "PROCESS TERMINATION" FLAG
002172' 052713 000010          BIS   #WT4IOT,(R3)    ;SET WAITING FOR I/O TERM FLAG
002176' 110114                 MOVB  R1,(R4)         ;ISSUE THE CMND
002200' 005767 175576          TST   DFLGWD          ;"NOWAIT" BIT SET?
002204' 100405                 BMI   WTNOT           ;N,Y-WTNOT
002206' 004577 175634          JSR   R5,@CIOBSY      ;WAIT FOR I/O TO COMPLETE
002212' 004767 001252    CMDEND: JSR   PC,PROCTM     ;GO PROCESS TERMINATION
002216' 000205           CMDEX:  RTS   R5             ;EXIT IN-LINE TO USER PROG

002220' 042713 000010    WTNOT:  BIC   #WT4IOT,(R3)  ;RESET WAITING FOR I/O TERM
002224' 000774                 BR    CMDEX           ;GO TO EXIT
```

```
                                        .SBTTL  TC11 NON-INTERRUPT COMMAND & DATA SERVICING

002226' 042701  000100          NONINT: BIC     #100,R1                 ;RESET INT ENABLE IN CMND
002232' 110114                          MOVB    R1,(R4)                 ;ISSUE RONUM/WRTH CMND
002234' 032702  000010                  BIT     #10,R2                  ;BLK SEARCH TYPE OF CMND?
002240' 001506                          BEQ     70$                     ;Y,N-70$
002242' 105714                  10$:    TSTB    (R4)                    ;READY SET?
002244' 100376                          BPL     10$                     ;Y,N-10$
002246' 012701  000003                  MOV     #3,R1                   ;SET UP RONUM CMND CODE
002252' 004767  000454                  JSR     PC,SEARCH               ;GO SEARCH FOR BLK
002256' 000430                          BR      25$                     ;ERROR ? (N,Y-25$)
002260' 032767  000010  176442          BIT     #10,CURFLG              ;FOUND THE BLK?
002266' 001365                          BNE     10$                     ;Y,N-10$
002270' 004767  000222                  JSR     PC,NINTSU               ;GO SET UP REGISTERS
002274' 032702  000200                  BIT     #200,R2                 ;"RDALL" CMND?
002300' 001443                          BEQ     50$                     ;Y,N-50$

                                        ;"RDALL" SERVICING

002302' 105714                  20$:    TSTB    (R4)                    ;READY SET?
002304' 100376                          BPL     20$                     ;Y,N-20$
002306' 005714                          TST     (R4)                    ;ERROR BIT SET?
002310' 100415                          BMI     30$                     ;N,Y-30$
002312' 016402  177776                  MOV     -2(R4),R2               ;GET TCST REG WITH BITS 16 & 17
002316' 042702  177774                  BIC     #177774,R2              ;RESET OTHER BITS
002322' 010220                          MOV     R2,(R0)+                ;STORE WORD WITH BITS 16 & 17
002324' 005201                          INC     R1                      ;DECR NEG WORD CNT
002326' 001411                          BEQ     40$                     ;CNT = 0? (N,Y-40$)
002330' 011320                          MOV     (R3),(R0)+              ;STORE BITS 0-15 FROM TCDT
002332' 005201                          INC     R1                      ;DECR NEG WORD CNT
002334' 001362                          BNE     20$                     ;CNT = 0? (Y,N-20$)
002336' 000405                          BR      40$                     ;GO TO TERMINATION

                                        ;NON-INT CMND TERMINATION

002340' 004767  000152          25$:    JSR     PC,NINTSU               ;SET UP CORRECT REG VALUES
002344' 052767  000001  175430  30$:    BIS     #1,DFLGWD               ;SET THE "ERROR" FLAG
002352' 010167  176362          40$:    MOV     R1,FINCNT               ;STORE FINAL WORD CNT
002356' 010164  000002                  MOV     R1,2(R4)                ;MOVE FINAL CNT TO TCWC
002362' 010064  000004                  MOV     R0,4(R4)                ;MOVE FINAL BUS ADR TO TCBA
002366' 004567  001610                  JSR     R5,STSTAT               ;STORE ALL STATUS REGISTERS
002372' 176260                          .WORD   ISTAT-.
002374' 005713                          TST     (R3)                    ;KNOCK DOWN READY IF STILL UP
002376' 112714  000011                  MOVB    #011,(R4)               ;ISSUE "STOP" CMND
002402' 004767  001550                  JSR     PC,SUPTAD               ;RESTORE PROG TBL ADR
002406' 000701                          BR      CMDEND                  ;GO PROCESS TERMINATION & EXIT

                                        ;"WRALL" SERVICING

002410' 105714                  50$:    TSTB    (R4)                    ;READY SET?
002412' 100376                          BPL     50$                     ;Y,N-50$
002414' 005714                          TST     (R4)                    ;ERROR BIT SET?
002416' 100752                          BMI     30$                     ;N,Y-30$
002420' 012002                          MOV     (R0)+,R2                ;GET WORD WITH BITS 16 & 17
002422' 042702  177774                  BIC     #177774,R2              ;RESET OTHER BITS
```

```
002426' 010264  177776          MOV     R2,-2(R4)       ;SET BITS 16 & 17 INTO TCST
002432' 005201                  INC     R1              ;DECR NEG WORD CNT
002434' 001403                  BEQ     60$             ;CNT = 0? (N,Y-60$)
002436' 012013                  MOV     (R0)+,(R3)      ;MOVE BITS 0-15 WORD TO TCDT
002440' 005201                  INC     R1              ;DECR NEG WORD CNT
002442' 001362                  BNE     50$             ;CNT = 0? (Y,N-50$)
002444' 032754  001000  177776  60$:    BIT     #1000,-2(R4)    ;DATA MISSED SET YET?
002452' 001774                  BEQ     60$             ;Y,N-60$
002454' 000736                  BR      40$             ;GO TO TERMINATION


                                ;"WRTM" SERVICING

002456' 004767  000034          70$:    JSR     PC,NINTSU       ;SET UP REGISTERS
002462' 105714                  80$:    TSTB    (R4)            ;READY SET YET?
002464' 100376                  BPL     80$             ;Y,N-80$
002466' 005714                  TST     (R4)            ;ERROR BIT SET?
002470' 100725                  BMI     30$             ;N,Y-30$
002472' 005701                  TST     R1              ;WORD CNT = 0?
002474' 001403                  BEQ     90$             ;N,Y-90$
002476' 012013                  MOV     (R0)+,(R3)      ;MOVE DATA WORD TO TCDT
002500' 005201                  INC     R1              ;DECR NEG WORD CNT
002502' 000767                  BR      80$             ;GO WAIT FOR NEXT WORD
002504' 012702  000310          90$:    MOV     #200.,R2        ;SET UP DELAY CNT
002510' 005302                  100$:   DEC     R2              ;DELAY FEW HUNDRED MICROSEC'S
002512' 001376                  BNE     100$
002514' 000716                  BR      40$             ;GO TO TERMINATION


                                ;NON-INT REGISTER SETUP S/R

002516' 016400  000004          NINTSU: MOV     4(R4),R0        ;MOVE BUS ADR TO R0
002522' 016401  000002          MOV     2(R4),R1        ;MOVE NEG WORD CNT TO R1
002526' 010403                  MOV     R4,R3           ;SET UP TCDT ADR IN R3
002530' 062703  000006          ADD     #6,R3
002534' 000207                  RTS     PC              ;EXIT IN-LINE
```

```
                                        .SBTTL  TC11 INTERRUPT SERVICE ROUTINE
                                
002536' 004067 001362           TCINT:  JSR     R0,SAVREG       ;SAVE ALL REGISTERS
002542' 005267 176152                   INC     INTCNT          ;ADD 1 TO INTERRUPT CNT
002546' 004767 001404                   JSR     PC,SUPTAD       ;SET UP PROG TBL & TCCM ADR'S
002552' 004567 001424                   JSR     R5,STSTAT       ;STORE ALL DEV REG'S
002556' 176074                          .WORD   ISTAT-.
002560' 016702 176144                   MOV     CURFLG,R2       ;GET THIS CMND'S FLGWD
002564' 032702 000010                   BIT     #10,R2          ;IN BLOCK SEARCH MODE?
002570' 001406                          BEQ     5$              ;Y,N-5$
002572' 012701 000103                   MOV     #103,R1         ;SET UP RONUM WITH INT ENABLE
002576' 004767 000130                   JSR     PC,SEARCH       ;GO SEARCH FOR SPECIFIED BLK
002602' 000403                          BR      10$             ;ERROR ON SEARCH? (N,Y-10$)
002604' 000446                          BR      60$             ;GO TO INT EXIT
002606' 005714                  5$:     TST     (R4)            ; IS THERE AN ERR CONDITION?
002610' 100006                          BPL     30$             ;Y,N-30$
002612' 052767 000001 175162    10$:    BIS     #1,DFLGWD       ;SET THE ERROR FLAG
002620' 112714 000011           20$:    MOVB    #011,(R4)       ;STOP TAPE & RESET INT ENABLE
002624' 000431                          BR      50$             ;GO TO CMND TERMINATION
                                
                                        ;"RONUM" USER CMND INT
                                
002626' 032702 000100           30$:    BIT     #100,R2         ;DOING A USER "RONUM" CMND?
002632' 001421                          BEQ     40$             ;Y,N-40$
002634' 016401 000006                   MOV     6(R4),R1        ;GET BLK # READ
002640' 016400 000004                   MOV     4(R4),R0        ;GET CURRENT DATA ADR
002644' 004777 175230                   JSR     PC,@PUTBYT      ;HAVE MPG STORE 1 BYTE
002650' 000301                          SWAB    R1              ;SET UP FOR 2ND BYTE
002652' 004777 175222                   JSR     PC,@PUTBYT      ;STORE IT TOO
002656' 010064 000004                   MOV     R0,4(R4)        ;STORE NEW DATA ADR
002662' 005264 000002                   INC     2(R4)           ;DECR NEG WORD CNT
002666' 001754                          BEQ     20$             ;CNT = 0? (N,Y-20$)
002670' 112714 000103                   MOVB    #103,(R4)       ;ISSUE "RONUM" CMND AGAIN
002674' 000412                          BR      60$             ;GO TO INTERRUPT EXIT
                                
                                        ;FINAL INTERRUPT PROCESSING
                                
002676' 032702 000040           40$:    BIT     #40,R2          ;ISSUE "STOP" CMND AT FINAL INT?
002702' 001746                          BEQ     20$             ;N,Y-20$
002704' 042714 000100                   BIC     #100,(R4)       ;RESET INT ENABLE
002710' 016467 000002 176022    50$:    MOV     2(R4),FINCNT    ;STORE FINAL WORD COUNT
002716' 042713 000010                   BIC     #WT4IOT,(R3)    ;RESET WAITING FOR I/O TERM
002722' 004067 001212           60$:    JSR     R0,RESREG       ;RESTORE ALL REGISTERS
002726' 000177 175142                   JMP     @RTNINT         ;EXIT FROM INTERRUPT
```

```
                                        .SBTTL  TC11 SEARCH FOR DECTAPE BLOCK S/R


                                        ;SEARCH FOR SPECIFIED BLOCK

                                                ;JSR    PC,SEARCH       S/R CALL
                                                ;BR     LABEL           EXECUTED IF AN ERROR

                                                ;R1 = RONUM CMND WITH OR W/O INT ENABLE
                                                ;R3 = PROG TBL ADR
                                                ;R4 = TCCM ADR
                                                ;DESTROYS R0,R1

002732' 116300 000035          SEARCH: MOVB    PCURDV(R3),R0           ;GET CURR DEV #
002736' 000300                         SWAB    R0                      ;ALIGN DEV # BITS
002740' 050001                         BIS     R0,R1                   ;SET DEV # INTO CMND
002742' 016700 175036                  MOV     BLK,R0                  ;GET DESIRED BLOCK #
002746' 005714                         TST     (R4)                    ;ERROR BIT SET?
002750' 100475                         BMI     110$                    ;N,Y-110$
002752' 032767 004000 175022           BIT     #4000,DFLGWD            ;DOING I/O'S IN REVERSE?
002760' 001046                         BNE     100$                    ;N,Y-100$

                                        ;SEARCH FOR BLK FOR FWD I/O

002762' 026467 000006 175746           CMP     6(R4),INBLKN            ;AT THE BLK WE'RE LOOKING FOR?
002770' 001430                         BEQ     90$                     ;N,Y-90$
002772' 002421                         BLT     80$                     ;PAST THE BLOCK? (Y,N-80$)
002774' 162700 000002          60$:    SUB     #2,R0                   ;SUB 2 FROM ORG BLK #
003000' 052701 004000                  BIS     #4000,R1                ;SET REV BIT IN CMND
003004' 032714 004000                  BIT     #4000,(R4)              ;ALREADY GOING REV?
003010' 001012                         BNE     80$                     ;N,Y-80$
003012' 005267 175724          70$:    INC     REVCNT                  ;INCR REVERSAL CNT
003016' 026727 175720 000006           CMP     REVCNT,#6               ;DONE 6 DIRECTION REVERSALS?
003024' 103404                         BLO     80$                     ;Y,N-80$
003026' 052767 000010 174746           BIS     #10,DFLGWD              ;SET BLK SRCH ERR FLG
003034' 000405                         BR      88$                     ;GO TO ERROR EXIT
003036' 010067 175674          80$:    MOV     R0,INBLKN               ;SAVE BLK # (ORG OR ADJUSTED)
003042' 010114                 85$:    MOV     R1,(R4)                 ;ISSUE DECTAPE CMND
003044' 062716 000002                  ADD     #2,(SP)                 ;SET UP NORMAL RET ADR
003050' 000207                 88$:    RTS     PC                      ;EXIT IN-LINE
003052' 032714 004000          90$:    BIT     #4000,(R4)              ;ARE WE GOING FWD?
003056' 001346                         BNE     60$                     ;Y,N-60$
003060' 105001                 95$:    CLRB    R1                      ;CLEAR CMND CODE FIELD
003062' 156701 175646                  BISB    CURCMD,R1               ;SET IN REQUESTED CMND CODE
003066' 042767 000010 175634           BIC     #10,CURFLG              ;RESET BLK SEARCH MODE FLAG
003074' 000762                         BR      85$                     ;GO ISSUE CMND

                                        ;SEARCH FOR BLK FOR REV I/O

003076' 052701 004000          100$:   BIS     #4000,R1                ;INITIALIZE TO REV DIRECTION
003102' 026467 000006 175626           CMP     6(R4),INBLKN            ;THIS THE BLOCK WE WANT?
003110' 001411                         BEQ     105$                    ;N,Y-105$
003112' 003351                         BGT     80$                     ;IN FRONT OF THE BLK? (Y,N-80$)
003114' 042701 004000          102$:   BIC     #4000,R1                ;SET DIR TO FWD
003120' 062700 000002                  ADD     #2,R0                   ;ALLOW FOR TURN AROUND
003124' 032714 004000                  BIT     #4000,(R4)              ;ALREADY GOING FWD?
```

```
003130' 001742                      BEQ     80$             ;N,Y-80$
003132' 000727                      BR      70$             ;GO CK REVERSAL CNT
003134' 032714  004000      105$:   BIT     #4000,(R4)      ;ARE WE GOING REV?
003140' 001347                      BNE     95$             ;N,Y-95$
003142' 000764                      BR      102$            ;GO STORE ADJUSTED BLK #

003144' 005764  177776      110$:   TST     -2(R4)          ;END ZONE?
003150' 100337                      BPL     88$             ;Y,N-88$
003152' 032714  004000              BIT     #4000,(R4)      ;GOING REV?
003156' 001315                      BNE     70$             ;N,Y-70$
003160' 052701  004000              BIS     #4000,R1        ;SET DIRECTION TO REV
003164' 000712                      BR      70$             ;GO CK REVERSAL CNT
```

```
                                                .SBTTL  SUBROUTINES FOR TC11 FUNCTION ROUTINES


                                                ;CHECK IF DEVICE IS BUSY AND WAIT IF IT IS

                                                        ;JSR    PC,CKDBSY       S/R CALL

                                                        ;DESTROYS R0,R3,R4
                                                        ;ON EXIT:
                                                        ;R3 = PROG TBL ADR
                                                        ;R4 = TCCM ADR

003166' 004767  000764          CKDBSY: JSR     PC,SUPTAD               ;SET UP PROG TBL & TCCM ADR'S
003172' 032714  000100          10$:    BIT     #100,(R4)               ;INT ENABLE ON?
003176' 001403                          BEQ     20$                     ;Y,N-20$
003200' 004577  174642                  JSR     R5,@CIOBSY              ;RELEASE CONTROL
003204' 000772                          BR      10$                     ;GO CK AGAIN
003206' 032767  000002  174566  20$:    BIT     #2,DFLGWD               ;HAVE TO PROCESS PREV TERMINATION?
003214' 001403                          BEQ     30$                     ;Y,N-30$
003216' 004767  000246                  JSR     PC,PROCTM               ;GO PROCESS TERMINATION
003222' 000763                          BR      10$                     ;GO RECHECK INT ENABLE
003224' 016767  174576  000012  30$:    MOV     IVCTAD,40$              ;STORE INT VECTOR ADR
003232' 016767  174572  000006          MOV     PSWD,45$                ;STORE PROC STATUS WORD
003240' 004577  174622                  JSR     R5,@SETVEC              ;GO SET UP THE VECTOR
003244' 000000                  40$:    .WORD   XXXX                    ;  INT VECTOR ADR
003246' 000000                  45$:    .WORD   XXXX                    ;  PSW
003250' 177266                          .WORD   TCINT-.                 ;  REL INT ROUT ADR
003252' 010567  175446                  MOV     R5,ERRADR               ;SAVE CURR USER STMNT ADR
003256' 162767  000004  175440          SUB     #4,ERRADR
003264' 000207                          RTS     PC                      ;EXIT IN-LINE


                                                ;ERROR INFORMATION DISPLAY S/R

                                                        ;JSR    R5,ERRCS        S/R CALL FOR CURR STATUS
                                                        ;JSR    R5,ERRIS        S/R CALL FOR INT STATUS
                                                        ;.WORD  MSGADR-ERMBAS   REL ADR OF ERROR MSG
                                                        ;.WORD  MSGCNT          # OF BYTES IN ERROR MSG
                                                        ;DESTROYS R0,R1,R2

003266' 004567  000710          ERRCS:  JSR     R5,STSTAT               ;STORE CURR STATUS
003272' 175372                          .WORD   CSTAT-.
003274' 012767  175262  000100  ERRCS1: MOV     #CSTAT-ERSTAD,ERSTAD    ;STORE ADR OF CURR STATUS
003302' 000403                          BR      ERRCOM                  ;GO TO COMMON POINT
003304' 012767  175250  000070  ERRIS:  MOV     #ISTAT-ERSTAD,ERSTAD    ;STORE ADR OF LAST INT STATUS
003312' 012567  000034          ERRCOM: MOV     (R5)+,ERMBAS            ;STORE MSG ADR
003316' 012567  000032                  MOV     (R5)+,ERMBAS+2          ;STORE MSG CNT
003322' 005267  175366                  INC     ERRCNT                  ;ADD 1 TO ERROR CNT
003326' 032763  020000  000002          BIT     #PRONER,POPSW(R3)       ;ERROR PRINTING INHIBITED?
003334' 001054                          BNE     ERREX                   ;N,Y-ERREX
003336' 010446                          MOV     R4,-(SP)                ;SAVE R4
003340' 005004                          CLR     R4                      ;SET USER MODE PRINT FLAG
003342' 004767  000660                  JSR     PC,DISUNM               ;DISPLAY UNIT #
003346' 004567  001060                  JSR     R5,PRINT                ;PRINT ERROR MSG SPECIFIED
003352' 000000                  ERMBAS: .WORD   XXXX
003354' 000000                          .WORD   XXXX
```

```
 922  003356' 026727 177770 001701          CMP     ERMBAS,#INVDVN-ERMBAS   ;INVALID UNIT # ERROR?
 923  003364' 001411                        BEQ     ERRSNM                  ;N,Y-ERRSNM
 924  003366' 026727 177760 001713          CMP     ERMBAS,#INVBKN-ERMBAS   ;INVALID BLOCK # ERROR?
 925  003374' 001403                        BEQ     ERPBKN                  ;N,Y-ERPBKN
 926  003376' 004567 000714                 JSR     R5,DISPST               ;DISPLAY STATUS REG'S
 927  003402' 000000                ERSTAD: .WORD   XXXX
 928  003404' 004767 000776         ERPBKN: JSR     PC,PRTIWD               ;DISPLAY CURR BLK #
 929  003410' 016300 000022         ERRSNM: MOV     PSRCST(R3),R0           ;GET ADR OF SRC STMNTS
 930  003414' 111001                10$:    MOVB    (R0),R1                 ;SAVE STMNT LENGTH
 931  003416' 026067 000004 175300         CMP     4(R0),ERRADR            ;ERROR OCCUR ON THIS STMNT?
 932  003424' 001402                        BEQ     20$                     ;N,Y-20$
 933  003426' 060100                        ADD     R1,R0                   ;POINT AT NXT STMNT
 934  003430' 000771                        BR      10$                     ;GO CK NXT STMNT
 935  003432' 005720                20$:    TST     (R0)+                   ;SET UP ADR OF STMNT # DATA
 936  003434' 010701                        MOV     PC,R1                   ;SET UP DATA OUTPUT ADR
 937  003436' 062701 001546                 ADD     #STMNUM-.,R1
 938  003442' 004577 174414                 JSR     R5,@DECASC              ;CONVERT IT TO ASCII
 939  003446' 012767 020040 001534         MOV     #20040,STMNUM+4         ;SET 2 LOW DIGITS TO SPACES
 940  003454' 004567 000752                 JSR     R5,PRINT                ;ISSUE STMNT # MSG
 941  003460' 001514                        .WORD   STMMSG-.
 942  003462' 177762                        .WORD   -14.
 943  003464' 012604                        MOV     (SP)+,R4                ;RESTORE R4
 944  003466' 000205                ERREX:  RTS     R5                      ;EXIT IN-LINE
 945
 946
 947                                        ;PROCESS TERMINATION OF PREVIOUS I/O FUNCTION
 948
 949                                                ;JSR    PC,PROCTM       S/R CALL
 950
 951  003470' 004067 000430         PROCTM: JSR     R0,SAVREG               ;SAVE ALL REG'S
 952  003474' 042767 000002 174300         BIC     #2,DFLGWD               ;RESET PROCESS TERMINATION FLAG
 953  003502' 032767 000004 175220         BIT     #4,CURFLG               ;INCR BYTE COUNT?
 954  003510' 001016                        BNE     6$                      ;Y,N-6$
 955  003512' 016700 175214                 MOV     CURCNT,R0               ;GET INITIAL WORD CNT
 956  003516' 016701 175216                 MOV     FINCNT,R1               ;GET FINAL WORD CNT
 957  003522' 100001                        BPL     2$                      ;IS IT NEGATIVE? (Y,N-2$)
 958  003524' 005401                        NEG     R1                      ;MAKE IT POSITIVE
 959  003526' 160100                2$:     SUB     R1,R0                   ;SUB REMAINING CNT FROM INITIAL CNT
 960  003530' 006300                        ASL     R0                      ;MAKE IT A BYTE CNT
 961  003532' 010067 174262                 MOV     R0,SIZE                 ;STORE # OF BYTES ACTUALLY XFERRED
 962  003536' 016701 175164                 MOV     CNTADR,R1               ;GET ADR OF BYTE CNT TOTALS
 963  003542' 060011                        ADD     R0,(R1)                 ;ADD IN THIS CNT
 964  003544' 005541                        ADC     -(R1)                   ;UPDATE MOST SIGNF WORD OF CNT
 965  003546' 032767 000001 174226  6$:     BIT     #1,DFLGWD               ;WAS THERE AN ERROR?
 966  003554' 001476                        BEQ     80$                     ;Y,N-80$
 967  003556' 012767 000001 174236         MOV     #1,ERR                  ;SET THE ERROR INDICATOR
 968  003564' 032763 000400 000002         BIT     #DOERCK,POPSW(R3)       ;SUPPOSED TO DO ERROR CHECKING?
 969  003572' 001065                        BNE     70$                     ;Y,N-70$
 970  003574' 032767 000010 174200         BIT     #10,DFLGWD              ;BLOCK SEARCH ERROR?
 971  003602' 001070                        BNE     90$                     ;N,Y-90$
 972  003604' 010701                        MOV     PC,R1                   ;GET ADR OF CODE  AREA IN ERR MSG
 973  003606' 062701 001422                 ADD     #CODFLD-.,R1
 974  003612' 010102                        MOV     R1,R2                   ;MOVE IT TO WORK REG
 975  003614' 012700 000023                 MOV     #19.,R0                 ;SET UP AREA SIZE
 976  003620' 112722 000040         10$:    MOVB    #40,(R2)+               ;CLEAR AREA TO SPACES
 977  003624' 005300                        DEC     R0
```

```
  978  003626' 001374                        BNE     10$
  979  003630' 010700                        MOV     PC,R0                   ;SET UP ADR OF ERROR CODE TBL
  980  003632' 062700  000144                ADD     #ERCDTB-.,R0
  981  003636' 010702                        MOV     PC,R2                   ;SET UP ADR OF STORED DEV REG'S
  982  003640' 062702  175013                ADD     #ISTAT+1-.,R2
  983  003644' 005046                        CLR     -(SP)                   ;INITIALIZE CODE CNT
  984  003646' 112004                20$:    MOVB    (R0)+,R4                ;GET ERROR BIT MASK CODE
  985  003650' 005704                        TST     R4                      ;END OF THE CODE TBL?
  986  003652' 001421                        BEQ     60$                     ;N,Y-60$
  987  003654' 130412                30$:    BITB    R4,(R2)                 ;THIS ERROR BIT SET IN STATUS BYTE
  988  003656' 001003                        BNE     40$                     ;N,Y-40$
  989  003660' 062700  000004                ADD     #4,R0                   ;POINT AT NXT CODE TBL ENTRY
  990  003664' 000770                        BR      20$                     ;GO CK FOR NXT CODE
  991  003666' 005716                40$:    TST     (SP)                    ;FIRST ERROR CODE IN MSG?
  992  003670' 001402                        BEQ     50$                     ;N,Y-50$
  993  003672' 112721  000054                MOVB    #',(R1)+                ;MOVE COMMA TO MSG
  994  003676' 005216                50$:    INC     (SP)                    ;INC # OF CODES IN THE MSG
  995  003700' 112021                        MOVB    (R0)+,(R1)+             ;MOVE ERROR CODE TO MSG
  996  003702' 112021                        MOVB    (R0)+,(R1)+
  997  003704' 112021                        MOVB    (R0)+,(R1)+
  998  003706' 112021                        MOVB    (R0)+,(R1)+
  999  003710' 022716  000004                CMP     #4,(SP)                 ;PUT 4 CODES IN THE MSG?
 1000  003714' 001354                        BNE     20$                     ;Y,N-20$
 1001  003716' 005726                60$:    TST     (SP)+                   ;RESTORE STACK
 1002  003720' 004567  177360                JSR     R5,ERRIS                ;GO ISSUE STATUS ERROR MSG
 1003  003724' 001640                        .WORD   TCEMSG-ERMBAS
 1004  003726' 000041                        .WORD   33.
 1005  003730' 004767  000114        65$:    JSR     PC,RINTV                ;GO RESET INT VECTOR
 1006  003734' 004067  000200                JSR     R0,RESREG               ;RESTORE REG'S
 1007  003740' 004577  174104                JSR     R5,@CUPGER              ;GO TO MPG ERR RETN POINT
 1008  003744' 000207                        RTS     PC                      ;EXIT IN-LINE
 1009  003746' 005267  174742        70$:    INC     ERRCNT                  ;ADD 1 TO ERROR CNT
 1010  003752' 004767  000072        80$:    JSR     PC,RINTV                ;GO RESET INT VECTOR
 1011  003756' 004067  000156                JSR     R0,RESREG               ;RESTORE REG'S
 1012  003762' 000207                        RTS     PC                      ;EXIT IN-LINE
 1013  003764' 004567  177314        90$:    JSR     R5,ERRIS                ;ISSUE BLK SRCH ERR MSG
 1014  003770' 001571                        .WORD   BSCHER-ERMBAS
 1015  003772' 000016                        .WORD   14.
 1016  003774' 000755                        BR      65$                     ;GO TO ERROR RETURN
 1017
 1018
 1019  003776' 042600  042116    132 ERCDTB: .ASCII  <200>/ENDZ/              ;ERROR MSG CODE TABLE
 1020  004003'    100  040520  042522         .ASCII  <100>/PARE/
 1021  004010' 046440  052113    105         .ASCII  <040>/MKTE/
 1022  004015'    020  046111  050117         .ASCII  <020>/ILOP/
 1023  004022' 051410  046105    105         .ASCII  <010>/SELE/
 1024  004027'    004  046102  046513         .ASCII  <004>/BLKM/
 1025  004034' 042002  052101    115         .ASCII  <002>/DATM/
 1026  004041'    001  042516  046530         .ASCII  <001>/NEXM/
 1027  004046'    000                        .BYTE   0                       ;TABLE TERMINATOR
 1028         004050'                        .EVEN
```

```
1030                                            ;RESET INTERRUPT VECTOR S/R
1031
1032                                                    ;JSR     PC,RINTV       S/R CALL
1033                                                    ;R3 MUST CONTAIN PROG TBL ADR
1034                                                    ;DESTROYS R0
1035
1036  004050' 004567  000020        RINTV:  JSR     R5,TVECT                ;GO CK IF I HAVE VECTOR CONTROL
1037  004054' 000406                BR      RINTEX                  ;BR IF I DON'T
1038  004056' 016767  173744  000004        MOV     IVCTAD,10$              ;GET CURR INT VECT ADR
1039  004064' 004577  174000                JSR     R5,@CLRVEC              ;GO HAVE MPG CLEAR IT
1040  004070' 000000                10$:    .WORD   XXXX
1041  004072' 000207                RINTEX: RTS     PC                      ;EXIT IN-LINE
1042
1043
1044                                            ;TEST INTERRUPT VECTOR S/R
1045
1046                                                    ;JSR     R5,TVECT       S/R CALL
1047                                                    ;BR      LABEL          EXECUTED IF NOT SAME
1048                                                    ;R3 MUST CONTAIN PROG TBL ADR
1049                                                    ;DESTROYS R0
1050
1051  004074' 01676   173726  000010  TVECT:  MOV     IVCTAD,20$              ;GET CURR INT VECT ADR
1052  004102' 016346  000004                MOV     PFWADR(R3),-(SP)        ;STORE FLGWD ADR TO IDENTIFY ME
1053  004106' 004577  173760                JSR     R5,@TSTVEC              ;DO I HAVE VECTOR CONTROL?
1054  004112' 000000                20$:    .WORD   XXXX                    ;  MPG WILL TELL ME SINCE I CAN'T
1055  004114' 176422                        .WORD   TCINT-.                 ;  GET AT LOWER MEM IF MEM MGMNT
1056  004116' 000401                BR      TVECTX                  ;BR IF I DONT'T HAVE CNTRL
1057  004120' 005725                TST     (R5)+                   ;BYPASS BR INST IN S/R CALL
1058  004122' 000205                TVECTX: RTS     R5                      ;EXIT IN-LINE
```

```
1060                                            .SBTTL  SUBROUTINES FOR TC11 DEVICE ROUTINE
1061
1062
1063
1064                                            ;SAVE REGISTERS R0 THRU R5
1065
1066                                                    ;JSR    R0,SAVREG       S/R CALL
1067
1068  004124' 010146                    SAVREG: MOV     R1,-(SP)                ;SAVE R0 THRU R5
1069  004126' 010246                            MOV     R2,-(SP)
1070  004130' 010346                            MOV     R3,-(SP)
1071  004132' 010446                            MOV     R4,-(SP)
1072  004134' 010546                            MOV     R5,-(SP)
1073  004136' 010007                            MOV     R0,PC                   ;EXIT IN-LINE
1074
1075
1076                                            ;RESTORE REGISTERS R0 THRU R5
1077
1078                                                    ;JSR    R0,RESREG       S/R CALL
1079
1080  004140' 005726                    RESREG: TST     (SP)+                   ;RESTORE R5 THRU R0
1081  004142' 012605                            MOV     (SP)+,R5
1082  004144' 012604                            MOV     (SP)+,R4
1083  004146' 012603                            MOV     (SP)+,R3
1084  004150' 012602                            MOV     (SP)+,R2
1085  004152' 012601                            MOV     (SP)+,R1
1086  004154' 000200                            RTS     R0                      ;EXIT IN-LINE
1087
1088
1089                                            ;SET PROGRAM'S PROG TABLE ADR IN R3 & TCCM ADR IN R4
1090
1091                                                    ;JSR    PC,SUPTAD       S/R CALL
1092
1093  004156' 010703                    SUPTAD: MOV     PC,R3                   ;SET UP LOCATION ZERO ADR
1094  004160' 062703  173620                    ADD     #LOCZ-.,R3
1095  004164' 166303  177776                    SUB     -2(R3),R3               ;SUBTRACT PROG TBL LENGTH
1096  004170' 016704  173630                    MOV     DREGAD,R4               ;GET DEV REG BASE ADR
1097  004174' 062704  000002                    ADD     #2,R4                   ;POINT AT TCCM
1098  004200' 000207                            RTS     PC                      ;EXIT IN-LINE
1099
1100
1101                                            ;STORE DEVICE'S STATUS REGISTERS
1102
1103                                                    ;JSR    R5,STSTAT       S/R CALL
1104                                                    ;.WORD  STADR-.         REL STORAGE ADR
1105                                                    ;DESTROYS R0,R1
1106
1107  004202' 010501                    STSTAT: MOV     R5,R1                   ;GET REL STORAGE ADR & MAKE
1108  004204' 062501                            ADD     (R5)+,R1                ;IT ABSOLUTE
1109  004206' 016700  173612                    MOV     DREGAD,R0               ;GET ADR OF DEV REG'S
1110  004212' 012021                            MOV     (R0)+,(R1)+             ;STORE ALL DEV REG'S
1111  004214' 012021                            MOV     (R0)+,(R1)+
1112  004216' 012021                            MOV     (R0)+,(R1)+
1113  004220' 012021                            MOV     (R0)+,(R1)+
1114  004222' 011011                            MOV     (R0),(R1)
1115  004224' 000205                            RTS     R5                      ;EXIT IN-LINE
```

```
                                        ;DISPLAY CURRENT UNIT #

                                                ;JSR     PC,DISUNM       S/R CALL
                                                ;R3 MUST CONTAIN PROG TBL ADR
                                                ;DESTROYS R0,R1,R2

004226' 012767  000031  000056  DISUNM: MOV     #25.,DISUML             ;INITIALIZE TO NORMAL MSG LNGTH
004234' 116300  000035                  MOVB    PCURDV(R3),R0           ;GET CURR UNIT #
004240' 020027  000007                  CMP     R0,#7                   ;VALID UNIT #?
004244' 101007                          BHI     DISUIV                  ;Y,N-DISUIV
004246' 004577  173606                  JSR     R5,@BTASLZ              ;CONVERT # TO DECIMAL ASCII
004252' 000410                          .WORD   UNASCI-.
004254' 016767  000406  000400          MOV     UNASCI+4,UNASCI         ;MOVE ASCII # TO 1ST TWO DIGITS
004262' 000410                          BR      DISUPR                  ;GO ISSUE MSG
004264' 012767  000035  000020  DISUIV: MOV     #29.,DISUML             ;SET UP ERR COND MSG LNGTH
004272' 042700  177400                  BIC     #177400,R0              ;RESET HIGH BYTE
004276' 004577  173554                  JSR     R5,@BINASC              ;CONVERT BINARY # TO ASCII
004302' 000360                          .WORD   UNASCI-.
004304' 004567  000122          DISUPR: JSR     R5,PRINT                ;GO ISSUE UNIT # MSG
004310' 000323                          .WORD   UNITMG-.
004312' 000031                  DISUML: .WORD   25.
004314' 000207                          RTS     PC                      ;EXIT IN-LINE


                                        ;TAILOR STATUS MSG & PRINT IT

                                                ;JSR     R5,DISPST       S/R CALL
                                                ;WORD    STATADR-.       REL ADR OF STATUS DATA
                                                ;DESTROYS R0,R1,R2

004316' 010502                  DISPST: MOV     R5,R2                   ;GET REL DATA ADR
004320' 062502                          ADD     (R5)+,R2                ;MAKE IT ABS
004322' 010701                          MOV     PC,R1                   ;SET UP ADR OF REG NAMES IN ASCII
004324' 062701  173572                  ADD     #DVREGS-.,R1
004330' 012746  000005                  MOV     #DVREGE-DVREGS/6,-(SP)  ;GET # OF REGISTERS TO DISPLAY
004334' 012167  000330          10$:    MOV     (R1)+,DVRGMG            ;MOVE REG NAME TO MSG
004340' 012167  000326                  MOV     (R1)+,DVRGMG+2
004344' 005721                          TST     (R1)+                   ;BYPASS DISP VALUE
004346' 012200                          MOV     (R2)+,R0                ;GET REG'S STORED VALUE
004350' 010146                          MOV     R1,-(SP)                ;SAVE R1 & R2
004352' 010246                          MOV     R2,-(SP)
004354' 004577  173476                  JSR     R5,@BINASC              ;CONVERT IT TO ASCII
004360' 000316                          .WORD   DVRGDT-.
004362' 004567  000044                  JSR     R5,PRINT                ;PRINT THE STATUS MSG
004366' 000302                          .WORD   DVRGMG-.
004370' 000014                          .WORD   12.
004372' 012602                          MOV     (SP)+,R2                ;RESTORE R1 & R2
004374' 012601                          MOV     (SP)+,R1
004376' 005316                          DEC     (SP)                    ;DECR REG CNT
004400' 001355                          BNE     10$                     ;DONE ALL? (Y,N-10$)
004402' 005726                          TST     (SP)+                   ;REMOVE COUNT FROM STACK
004404' 000205                          RTS     R5                      ;EXIT IN-LINE
```

```
                                        ;PRINT CURRENT "BLK" VALUE
                                        ;
                                        ;JSR    PC,PRTIWD       S/R CALL
                                        ;DESTROYS R0,R1,R2
                                        ;
004406' 016700 173372           PRTIWD: MOV     BLK,R0          ;GET BLK VALUE
004412' 004577 173440                   JSR     R5,@BINASC      ;CONVERT IT TO ASCII
004416' 000550                          .WORD   INFOBK-.
004420' 004567 000006                   JSR     R5,PRINT        ;ISSUE BLOCK # MSG
004424' 000535                          .WORD   INFOMG-.
004426' 000013                          .WORD   11.
004430' 000207                          RTS     PC              ;EXIT IN-LINE
                                        ;
                                        ;ISSUE MSG TO LIST DEVICE
                                        ;
                                        ;JSR    R5,PRINT        S/R CALL
                                        ;.WORD  MSGADR-.        REL ADR OF MSG
                                        ;.WORD  BYTCNT          MSG BYTE CNT (IF NEGATIVE,
                                        ;                         RESET PRT DEV DEDICATED.)
                                        ;R3 = PROG TBL ADR
                                        ;R4 = FLAGWORD -- IF NEGATIVE, USE CMND MODE PRINT
                                        ;DESTROYS R0,R1,R2
                                        ;
004432' 010500                  PRINT:  MOV     R5,R0           ;GET MSG ADR & MAKE IT ABS
004434' 062500                          ADD     (R5)+,R0
004436' 012501                          MOV     (R5)+,R1        ;GET BYTE COUNT
004440' 005704                          TST     R4              ;USE CMND MODE PRINT?
004442' 100030                          BPL     40$             ;Y,N-40$
004444' 010702                          MOV     PC,R2           ;SET UP LINK INFO ADR
004446' 062702 000040                   ADD     #2C[-.,R2
004452' 160200                          SUB     R2,R0           ;MAKE MSG ADR REL
004454' 010022                          MOV     R0,(R2)+        ;STORE MSG ADR
004456' 010112                          MOV     R1,(R2)         ;STORE MSG'S BYTE COUNT
004460' 100001                          BPL     10$             ;CNT NEG? (Y,N-10$)
004462' 005412                          NEG     (R2)            ;MAKE IT POSITIVE
004464' 016367 000006 000056    10$:    MOV     PRGCIN(R3),PROGNM       ;STORE PROG'S # IN MSG
004472' 004577 173356                   JSR     R5,@CLIST       ;ISSUE PROG #
004476' 000050                          .WORD   PNMMSG-.
004500' 000005                          .WORD   5
004502' 004577 173346                   JSR     R5,@CLIST       ;ISSUE MSG SPECIFIED
004506' 000000                  20$:    .WORD   XXXX
004510' 000000                          .WORD   XXXX
004512' 004577 173336                   JSR     R5,@CLIST       ;ISSUE A <CR> & <LF>
004516' 000240                          .WORD   CRLF-.
004520' 000002                          .WORD   2
004522' 000410                          BR      PRTEX           ;GO TO EXIT
004524' 010067 000010           40$:    MOV     R0,50$          ;STORE MSG'S ABS ADR
004530' 010167 000006                   MOV     R1,60$          ;STORE ITS BYTE CNT
004534' 004577 173312                   JSR     R5,@ULIST       ;GO TO MPG TO ISSUE THE MSG
004540' 000000                  50$:    .WORD   XXXX
004542' 000000                  60$:    .WORD   XXXX
004544' 000205                  PRTEX:  RTS     R5              ;EXIT IN-LINE
```

```
                                                .SBTTL  TC11 MESSAGE STORAGE AREA
                                                .NLIST  BEX
                                                .EVEN
005546' 021520                          PNMMSG: .ASCII  /P#/
005550' 054130      011                 PROGNM: .ASCII  /XX/<011>
004553'    101  020124  040514  ATIMSG: .ASCII  'AT LAST INT/NON-INT TERM:'
004604' 052503  051122  047105  CURMSG: .ASCII  /CURRENTLY:/
004616' 047105  020104  043117  RENDMG: .ASCII  /END OF REPORT/
                                        .QUO
004633'    052  025052  052052  UNITMG: .ASCII  /####TC11 DECTAPE UNIT: /
004662' 054130  054130  054130  UNASCI: .ASCII  /XXXXXX/
                                        .EVEN
004670' 054130  054130  020075  DVRGMG: .ASCII  /XXXX= /
004676' 054130  054130  054130  DVRGDT: .ASCII  /XXXXXX/
004704' 054502  042524  036123  CNTSMG: .ASCII  /BYTES:  RD= /
004720' 054130  054130  054130  BCHRD:  .ASCII  /XXXXXXXXXXXX  WR= /
004742' 054130  054130  054130  BCHWR:  .ASCII  /XXXXXXXXXXXX/
005756' 005015                  CRLF:   .ASCII  <015><012>
005760' 041411  047115  051504          .ASCII  <011>/CMNDS:  RD= /
005775'    130  054130  054130  CMDCRD: .ASCII  /XXXXXX WR= /
006010' 054130  054130  054130  CMDCHR: .ASCII  /XXXXXX MISC= /
006025'    130  054130  054130  CMDCMS: .ASCII  /XXXXXX/<015><012>
006036'    011  051105  047522          .ASCII  <011>/ERRORS:  DEV= /
006054' 054130  054130  054130  CNTERR: .ASCII  /XXXXXX DATA= /
006071'    130  054130  054130  CNTDER: .ASCII  /XXXXXX/<015><012>
006101'    011  047111  042524          .ASCII  <011>/INTERRUPTS:  /
006117'    130  054130  054130  CNTINT: .ASCII  /XXXXXX/
           006125'                  CNTSEN= .
006125'    124  046511  047505  IOTO:   .ASCII  'TIMEOUT ON I/O'
006143'    102  045514  051440  BSCHER: .ASCII  /BLK SEARCH ERR/
006161'    102  045514  020075  INFOMG: .ASCII  /BLK= /
006166' 054130  054130  054130  INFOBK: .ASCII  /XXXXXX/
                                        .EVEN
005174' 052123  047115  020124  STMNMG: .ASCII  /STMNT # /
005204' 054130  054130  054130  STMNUM: .ASCII  /XXXXXX/
005212' 052123  052101  051525  TCEMSG: .ASCII  /STATUS ERROR: /
005230' 000023                  CODFLD: .BLKB   19.
005253'    111  053116  052440  INVDVN: .ASCII  /INV UNIT #/
005265'    111  053116  041040  INVBKN: .ASCII  /INV BLK #/
                                        .EVEN

                                                .LIST   BEX

           005276'                  DVREND= .
```

```
                                .SBTTL  FORMATS FOR PROGRAM & DEVICE ROUTINE TABLES

                        ;       PROGRAM TABLE FORMAT

000242                  PTLGTH= 162.    ;PROGRAM TABLE LENGTH - NON MEM MGMNT VERSION OF MPG

                        ;(PTLGTH= 212.  ;PROGRAM TABLE LENGTH - MEM MGMNT VERSION OF MPG)

000000                  PFLGWD= +0.     ;PROGRAM FLAG WORD - 1 WORD

000002                          URSTOP= 2       ; 1 = USER HAS STOPPED THIS PROGRAM
000004                          ERSTOP= 4       ; 1 = AN ERROR HAS STOPPED THIS PROGRAM
000010                          WT4IOT= 10      ; 1 = WAITING FOR I/O TERMINATION
000020                          CTPRIO= 20      ; 1 = CONSOLE OR PRINTER I/O IN PROGRESS
000040                          SETDED= 40      ; 1 = THIS PROG SET THE PRT DEV DEDICATED FLAG
000100                          OCPRES= 100     ; 1 = OBJ CODE IS PRESENT
000200                          USEUBM= 200     ; 1 = THIS PROG USES THE UNIBUS MAP (MEM MGMNT ONLY
100000                          ACTIVE= 100000  ; 1 = PROGRAM IS ACTIVE (SPECIFIED FOR EXECUTION)

000002                  POPSW=  +2.     ;PROGRAM'S OPERATION SWITCHES - 1 WORD

100000                          STONER= 100000  ; 1 = STOP PROG EXECUTION UPON ERROR
040000                          CYCPRG= 40000   ; 1 = CYCLE PROGRAM (ON CURRENT DEVICE)
020000                          PRONER= 20000   ; 1 = DO NOT PRINT ON ERROR
010000                          BIT12=  10000   ; 0 = NOT USED
004000                          BIT11=  4000    ; 0 = NOT USED
002000                          CYCDVL= 2000    ; 1 = CYCLE THE DEVICE LIST
001000                          GTNXTD= 1000    ; 1 = CYCLE ON SAME DEVICE UPON ERROR
000400                          DOERCK= 400     ; 1 = DON'T DO ERROR CHECKING
000200                          SPOPER= 200     ; 1 = DEVICE SPECIAL OPERATION
000100                          BIT6=   100     ; 0 = NOT USED
000040                          DOIOT=  40      ; 1 = DO NOT PERFORM I/O TIMEOUT
000020                          AUTORP= 20      ; 1 = DO NOT AUTOMATICALLY DISPLAY COUNTS
000010                          AURPEP= 10      ; 1 = AUTO DISPLAY COUNTS AT END OF FINAL PASS ONLY
000004                          HSKPEP= 4       ; 1 = HOUSEKEEP COUNTS ONLY AT RUN COMMAND
000002                          PFBBOV= 2       ; 1 = PRINT FIRST BAD BYTE ONLY ON VERIFY
000001                          NOCOMP= 1       ; 1 = DO NOT PRINT PROG COMPLETED MSG

000004                  PFWADR= +4.     ;*;PROGRAM FLAGWORD ADDRESS - 1 WORD

000006                  PASCIN= +6.     ;PROGRAM'S NUMBER IN ASCII - 1 WORD

000010                  PNAME=  +8.     ;PROGRAM'S NAME IN ASCII - 6 BYTES

000016                  PRDIOA= +14.    ;ADDRESS OF READ I/O AREA - 1 WORD

000020                  PWRIOA= +16.    ;ADDRESS OF WRITE I/O AREA - 1 WORD

000022                  PSRCST= +18.    ;SOURCE STATEMENTS START ADDRESS - 1 WORD

000024                  POBJST= +20.    ;OBJECT CODE START ADDRESS - 1 WORD

000026                  PLNGTH= +22.    ;PROG AREA LENGTH (OBJ END MINUS PROG TBL START) - 1 WORD

000030                  PTOCNT= +24.    ;I/O TIMEOUT COUNT - 1 WORD
```

```
1329    000032                  PMDLCD= +26.    ;DEV ROUT MODEL # CODE - 1 WORD
1330
1331    000034                  PDPNTR= +28.    ;CURRENT DEVICE NUMBER POINTER - 1 BYTE
1332
1333    000035                  PCURDV= +29.    ;CURRENT DEVICE # - 1 BYTE
1334
1335    000036                  PDNUMS= +30.    ;DEVICE NUMBERS - 16 BYTES
1336
1337    000056                  PTEM0=  +46.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1338
1339    000060                  PTEM1=  +48.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1340
1341    000062                  PTEM2=  +50.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1342
1343    000064                  PTEM3=  +52.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1344
1345    000066                  PTEM4=  +54.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1346
1347    000070                  PTEM5=  +56.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1348
1349    000072                  PTEM6=  +58.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1350
1351    000074                  PTEM7=  +60.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1352
1353    000076                  PTEM8=  +62.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1354
1355    000100                  PTEM9=  +64.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1356
1357    000102                  PTEM10= +66.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1358
1359    000104                  PTEM11= +68.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1360
1361    000106                  PTEM12= +70.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1362
1363    000110                  PTEM13= +72.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1364
1365    000112                  PTEM14= +74.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1366
1367    000114                  PTEM15= +76.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1368
1369    000116                  PNBR=   +78.    ;NUMBER OF BYTES TO TRANSFER ON MOVE (NBR) - 1 WORD
1370
1371    000120                  PSRC=   +80.    ;DATA SOURCE ADDRESS ON MOVE (SRC) - 1 WORD
1372
1373    000122                  PDST=   +82.    ;DATA DESTINATION ADDRESS ON MOVE (DST) - 1 WORD
1374
1375    000124                  PSTKCT= +84.    ;# OF WORDS (X 2) SAVED OFF STACK - 1 WORD
1376
1377    000126                  PSTKSV= +86.    ;STACK WORDS STORAGE AREA - 30 WORDS
1378
1379    000222                  PSVREG= +146.   ;USER'S R0 THRU R5 REGISTERS STORAGE AREA - 6 WORDS
1380
1381    000236                  PUSRPC= +158.   ;USER'S CURRENT PROGRAM COUNTER - 1 WORD
1382
```

```
1384                                    ;FOLLOWING ENTRIES (PRDIOX THRU PUBMAP) ARE ONLY IN MEM MGMNT VERSION
1385
1386                                ;(PRDIOX= +160. ;18/22 BIT ABSOLUTE ADDRESS OF READ I/O AREA - 2 WORDS)
1387                                ;(PRDIOV= +164. ;18 BIT VIRTUAL ADDRESS OF READ I/O AREA - 2 WORDS)
1388
1389                                ;(PWRIOX= +168. ;18/22 BIT ABSOLUTE ADDRESS OF WRITE I/O AREA - 2 WORDS)
1390                                ;(PWRIOV= +172. ;18 BIT VIRTUAL ADDRESS OF WRITE I/O AREA - 2 WORDS)
1391
1392                                ;(PUPARS= +176  ;STORAGE AREA FOR USER'S PAR'S 0 THRU 7 - 8 WORDS)
1393                                ;(PUPDRS= +192. ;STORAGE AREA FOR USER'S PDR'S 0 THRU 7 - 8 WORDS)
1394
1395                                ;(PUBMAP= +208. ;1ST UNIBUS MAP REG # AND # OF REGS USED - 1 WORD)
1396
1397                                        ;END OF MEM MGMNT ONLY ENTRIES
1398
1399    000240                      PTSIZE= +160.   ;PROGRAM TABLE SIZE IN BYTES - 1 WORD - NON MEM MGMNT
1400                                ;(PTSIZE= +210. ;PROGRAM TABLE SIZE IN BYTES - 1 WORD - MEM MGMNT VERSION)
1401
1402    000242                      PTEND=  +162.   ;END OF PROGRAM TABLE - NON MEM MGMNT VERSION
1403                                ;(PTEND=  +212. ;END OF PROGRAM TABLE - MEM MGMNT VERSION)
1404
```

```
                                ;       DEVICE ROUTINE TABLE
                                ;
000116                          DRTLTH= 78.     ;DEVICE ROUTINE TABLE LENGTH
                                ;
000000                          DEVRSZ= +0.     ;DEVICE ROUTINE SIZE IN BYTES - 1 WORD
000002                          DEVFWD= +2.     ;DEVICE ROUTINE FLAGWORD - 1 WORD
000004                          DEVIW1= +4.     ;DEVICE INTERFACE WORD # 1 - 1 WORD
000006                          DEVIW2= +6.     ;DEVICE INTERFACE WORD # 2 - 1 WORD
000010                          DEVIW3= +8.     ;DEVICE INTERFACE WORD # 3 - 1 WORD
000012                          DEVIW4= +10.    ;DEVICE INTERFACE WORD # 4 - 1 WORD
000014                          DEVIW5= +12.    ;DEVICE INTERFACE WORD # 5 - 1 WORD
000016                          DEVIW6= +14.    ;DEVICE INTERFACE WORD # 6 - 1 WORD
000020                          DEVIW7= +16.    ;DEVICE INTERFACE WORD # 7 - 1 WORD  (SIZE)
000022                          DEVIW8= +18.    ;DEVICE INTERFACE WORD # 8 - 1 WORD  (ERR)
000024                          DEVDRA= +20.    ;DEVICE REGISTERS ADDRESS - 1 WORD
000026                          DEVIVA= +22.    ;DEVICE INTERRUPT VECTOR ADDRESS - 1 WORD
000030                          DEVRPS= +24.    ;DEVICE READ PROCESSOR STATUS WORD (BUS REQ) - 1 WORD
000032                          DEVWPS= +26.    ;DEVICE WRITE PROC STATUS WORD (BUS REQ) - 1 WORD
000034                          DHKPAD= +28.    ;DEVICE ROUT HOUSEKEEPING ROUT REL ENTRY ADR - 1 WORD
000036                          DERPAD= +30.    ;DEVICE ROUT REPORT ROUT REL ENTRY ADR - 1 WORD
000040                          DKILAD= +32.    ;DEVICE ROUT KILL ROUTINE REL ENTRY ADR - 1 WORD
000042                          DECTAD= +34.    ;DEVICE ROUT ERROR COUNTER REL ADR - 1 WORD
000044                          DTOEAD= +36.    ;DEVICE ROUT TIMEOUT ERR ROUT REL ENTRY ADR - 1 WORD
000046                          DEVIOB= +38.    ;DEVICE I/O BUSY BRANCH ADDRESS (CIOBSY) - 1 WORD
000050                          DEVDER= +40.    ;DEVICE ERROR BRANCH ADDRESS (CUPGER) - 1 WORD
000052                          DVUPRT= +42.    ;USER MODE PRINT BRANCH ADDRESS (ULIST) - 1 WORD
000054                          DVCPRT= +44.    ;CMND MODE PRINT BRANCH ADDRESS (CLIST) - 1 WORD
000056                          DEVBTA= +46.    ;CONVERT BINARY TO ASCII BR ADR (BINASC) - 1 WORD
000060                          DVBTDA= +48.    ;CONVERT BINARY TO DECIMAL ASCII BR ADR (BTASL2) - 1 WORD
```

```
1466    000062                  DVPDTA= +50.    ;CONVERT PACKED DECIMAL TO ASCII BR ADR (DECASC) - 1 WORD
1467
1468    000064                  DVSFWD= +52.    ;MPG SYSTEM FLAGWORD ADDRESS (CSYSFW) - 1 WORD
1469
1470    000066                  DVSVEC= +54.    ;SET INTERRUPT VECTOR BR ADR (SETVEC) - 1 WORD
1471
1472    000070                  DVCVEC= +56.    ;CLEAR INTERRUPT VECTOR BR ADR (CLRVEC) - 1 WORD
1473
1474    000072                  DVTVEC= +58.    ;TEST INTERRUPT VECTOR BR ADR (TSTVEC) - 1 WORD
1475
1476    000074                  DVRINT= +60.    ;RETURN FROM INTERRUPT BR ADR (RTNINT) - 1 WORD
1477
1478    000076                  DVGETB= +62.    ;GET DATA BYTE BR ADR (GETBYT) - 1 WORD
1479
1480    000100                  DVPUTB= +64.    ;PUT DATA BYTE BR ADR (PUTBYT) - 1 WORD
1481
1482    000102                  DEVSTP= +66.    ;DEVICE ROUT REL SYMBOL TABLE POINTER - 1 WORD
1483
1484    000104                  DEVETP= +68.    ;DEVICE ROUT REL ENTRY TABLE POINTER - 1 WORD
1485
1486    000106                  DVPTEP= +70.    ;PACK TABLE EXTEN. REL POINTER - 1 WORD
1487
1488    000110                  DVVTEP= +72.    ;VECTOR TABLE EXTEN. REL POINTER - 1 WORD
1489
1490    000112                  DVCTEP= +74.    ;COMPILER TBL EXTEN. REL POINTER - 1 WORD
1491
1492    000114                  DVIWSP= +76.    ;DEVICE INTERFACE WORD SYMBOL TBL REL POINTER - 1 WORD
1493
1494    000116                  DRTEND= +78.    ;END OF DEVICE ROUTINE TABLE
1495
1496
1497    000001                          .END
```

```
ACTIVE= 100000
ATIMSG  004553R     002
AURPEP= 000010
AUTORP= 000020
BCMRD   004720R     002
BCMWR   004742R     002
BINASC  000056R     002
BIT11 = 004000
BIT12 = 010000
BIT6  = 000100
BLK     000004R     002
BSCHER  005143R     002
BTASLZ  000060R     002
BYRD    000676R     002
BYWR    000702R     002
CIOBSY  000046R     002
CKOBSY  003166R     002
CLIST   000054R     002
CLRVEC  000070R     002
CMDCMS  005025R     002
CMDCOM  001702R     002
CMDCRD  004775R     002
CMDCWR  005010R     002
CMDEND  002212R     002
CMDEX   002216R     002
CNTAOR  000726R     002
CNTDER  005071R     002
CNTERR  005054R     002
CNTINT  005117R     002
CNTSEN= 005125R     002
CNTSMG  004704R     002
CODFLD  005230R     002
CRLF    004756R     002
CSTAT   000664R     002
CSYSFW  000064R     002
CTPRIO= 000020
CUPGER  000050R     002
CURCMD  000734R     002
CURCNT  000732R     002
CURFLG  000730R     002
CURMSG  004604R     002
CYCOVL= 002000
CYCPRG= 040000
DATAER  000716R     002
DECASC  000062R     002
DECTAD= 000042
DERPAD= 000036
DEVBTA= 000056
DEVDER= 000050
DEVDRA= 000024
DEVETP= 000104
DEVFWD= 000002
DEVIOB= 000046
DEVIVA= 000026
DEVIW1= 000004
DEVIW2= 000006
DEVIW3= 000010
DEVIW4= 000012
DEVIW5= 000014
DEVIW6= 000016
DEVIW7= 000020
DEVIWB= 000022
DEVMPS= 000030
DEVRSZ= 000000
DEVSTP= 000102
DEVWPS= 000032
DFLGWD  0000002R    002
DHKPAD= 000034
DISCNT  001160R     002
DISCT1  001164R     002
DISPST  004316R     002
DISUIV  004264R     002
DISUML  004312R     002
DISUNM  004226R     002
DISUPR  004304R     002
DKILAD= 000040
DOERCK= 000400
DOIOT = 000040
DREGAD  000024R     002
DRTEND= 000116
DRTLTH= 000116
DTOEAD= 000044
DVBTDA= 000060
DVCMDS  000154R     002
DVCPRT= 000054
DVCPTE  000466R     002
DVCT. = 000112
DVCVEC= 000070
DVGETB= 000076
DVIWSP= 000114
DVIWST  000622R     002
DVMVTE  000406R     002
DVPDTA= 000062
DVPKTE  000246R     002
DVPTEP= 000106
DVPUTB= 000100
DVREGE= 000154R     002
DVREGS  000116R     002
DVREND= 005276R     002
DVREX   001264R     002
DVRGDT  004676R     002
DVRGMG  004670R     002
DVRINT= 000074
DVSFWD= 000064
DVSVEC= 000066
DVTVEC= 000072
DVUPRT= 000052
DVVTEP= 000110
ERCDTB  003776R     002
ERMBAS  003352R     002
ERPBKN  003404R     002
ERR     000022R     002
ERRADR  000724R     002
ERRCNT  000714R     002
ERRCOM  003312R     002
ERRCS   003266R     002
ERRCS1  003274R     002
ERREX   003466R     002
ERRTS   003304R     002
ERRSNM  003410R     002
ERSTAD  003402R     002
ERSTOP= 000004
FINCNT  000740R     002
FUNCEX  001474R     002
FWD     001476R     002
GETBYT  000076R     002
GTNXTD= 001000
HSKEEP  000744R     002
HSKPEN= 000744R     002
HSKPEP= 000004
HSKPST= 000652R     002
INBUKN  000736R     002
INFOBK  005166R     002
INFOMG  005161R     002
INTCNT  000720R     002
INVBKN  005265R     002
INVDVN  005253R     002
IOTO    005125R     002
ISTAT = 000652R     002
IVCTAD  000026R     002
KILL    001420R     002
KILLEX  001444R     002
LCOUNT  000632R     002
LFWD    000632R     002
LNWAIT  000632R     002
LOCZ    000000R     002
LROALL  000633R     002
LRONUM  000633R     002
LREV    000632R     002
LSTALL  000632R     002
LSTATS  000632R     002
LSTOP   000632R     002
LWAIT   000632R     002
LWRALL  000642R     002
LWRTM   000642R     002
MISCNT  000712R     002
MISCOM  001656R     002
NINTSU  002516R     002
NOCOMP= 000001
NONINT  002226R     002
NOWAIT  001466R     002
OCPRES= 000100
PASCIN= 000006
PC      =%0000007
PCURDV= 000035
PDNUMS= 000036
PDPNTR= 000034
PDST  = 000122
PFBBOV= 000002
PFLGWD= 000000
PFWADR= 000004
PLNGTH= 000026
PMDLCD= 000032
PNAME = 000010
PNBR  = 000116
PNMMSG  004546R     002
POBJST= 000024
POPSW = 000002
PROIOA= 000016
PRINT   004432R     002
PROCTM  003470R     002
PROGNM  004550R     002
PRONER= 020000
PRTEX   004544R     002
PRTIWD  004406R     002
PSRC  = 000120
PSRCST= 000022
PSTKCT= 000124
PSTKSV= 000126
PSVREG= 000222
PSWD    000030R     002
PTEM0 = 000056
PTEM1 = 000060
PTEM10= 000102
PTEM11= 000104
PTEM12= 000106
PTEM13= 000110
PTEM14= 000112
PTEM15= 000114
PTEM2 = 000062
PTEM3 = 000064
PTEM4 = 000066
PTEM5 = 000070
PTEM6 = 000072
PTEM7 = 000074
PTEM8 = 000076
PTEM9 = 000100
PTEND = 000242
PTLGTH= 000242
PTOCNT= 000030
PTSIZE= 000240
PUSRPC= 000236
PUTBYT  000100R     002
PWRIOA= 000020
RDALL   001610R     002
RDCNT   000706R     002
RDCOM   001526R     002
RDNUM   001576R     002
READ    001516R     002
RENDMG  004616R     002
REPORT  001000R     002
REPTBL  001274R     002
RESREG  004140R     002
REV     001506R     002
REVCNT  000742R     002
RINTEX  004072R     002
RINTV   004050R     002
RPTBAS  001230R     002
RPTEND  001254R     002
RPTLP   001212R     002
RTNINT  000074R     002
R0      =%0000000
R1      =%0000001
R2      =%0000002
R3      =%0000003
R4      =%0000004
R5      =%0000005
SAVREG  004124R     002
SEARCH  002732R     002
SETDED= 000040
SETVEC  000066R     002
SIZE    000020R     002
SP      =%0000006
SPOPER= 000200
STMNMG  005174R     002
STMNUM  005204R     002
STONER= 100000
STOP    001646R     002
STPALL  001670R     002
STSTAT  004202R     002
SUPTAD  004156R     002
TCEMSG  005212R     002
TCINT   002536R     002
TOECNT  000722R     002
TOUTER  001320R     002
TOUTEX  001416R     002
TSTVEC  000072R     002
TVECT   004074R     002
TVECTX  004122R     002
ULIST   000052R     002
UNASCI  004662R     002
UNITMG  004633R     002
URSTOP= 000002
USEUBM= 000200
WAIT    001446R     002
```

```
MAINDEC-11-DTTCA-B  TC11/TU56 DEVICE ROUTINE FOR MPG    MACY11 27(732)  24-SEP-76  14:00  PAGE 13-1                    SEQ 0388
DTTCAB.P11      SYMBOL TABLE

WRALL   001622R     002   WRCOM   001556R     002   WRTM    001634R     002   WT4IOT= 000010          .     = 005276R     002

. ABS.  000000      000
        000000      001
TC11    005276      002


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 *,DTTCAB/NL:TOC/DOC=DTTCAB.P11
 RUN-TIME: 4 9 .9 SECONDS
 RUN-TIME RATIO: 171/14=11.5
 CORE USED:  5K  (9 PAGES)

 DOCUMENT PAGES:  35
```