

MDBG IDENT H18 2/14/68

\*  
\*  
\*

SYSTEM FLAGS, OPDS, PARAMETERS, AND MACROS

\* ASSEMBLY FLAGS

|        |     |    |  |
|--------|-----|----|--|
| CPXF   | EQU | -1 | (NO CARD PUNCH)                        |
| CRXF   | EQU | 1  | (CARD READER)                          |
| PNXF   | EQU | 1  | (PAPER TAPE PUNCH)                     |
| LPXF   | EQU | -1 | (NO LINE PRINTER)                      |
| RELCHN | EQU | -1 | (OLD DRUM CHANNEL)                     |
| ARMF   | EQU | 1  | (ARMING FEATURE)                       |
| DEBUG  | EQU | 1  | (1 ASSEMBLE DEBUG PACKAGE, -1 DONT)    |
| COM    | EQU | 1  | (1 COMSHARE SYSTEM, -1 NOT)            |
| BITMAP | EQU | 1  | (1 FOR COMSHARE EXEC, -1 FOR BERKELEY) |
| RANFIL | EQU | -1 | (NO RANDOM FILE LOGIC)                 |
| NOLIST |     |    |  |

\* OPDS

|        |       |               |                                 |
|--------|-------|---------------|---------------------------------|
| SBRS   | OPD   | 17300000B,1,1 | SYSTEM MODE BRS                 |
| TSN    | OPD   | 00222000B,2   | GO FROM NORMAL TO MONITOR MODE  |
| CKN    | OPD   | 00220100B,2   | TURN ON THE CLOCK               |
| CKF    | OPD   | 00220200B,2   | TURN OFF THE CLOCK              |
| LRR1   | OPD   | 00220400B,2   | LOAD RELABELLING REGISTER 1     |
| LRR2   | OPD   | 00221000B,2   | LOAD RELABELLING REGISTER 2     |
| LRR3   | OPD   | 00221400B,2   | LOAD RELABELLING REGISTER 3     |
| TTYS   | MACRO |               |                                 |
|        | DATA  | 20277777B     |                                 |
|        | ENDM  |               |                                 |
| RSR    | MACRO |               |                                 |
|        | DATA  | 04050026B     |                                 |
|        | ENDM  |               | SKIP READY                      |
| RSE    | MACRO |               |                                 |
|        | DATA  | 04051026B     |                                 |
|        | ENDM  |               | SKIP NO ERROR                   |
| ADA    | MACRO | D             |                                 |
|        | EOM   | 10026B        |                                 |
|        | ENDM  |               |                                 |
| DET    | MACRO | D             |                                 |
|        | SKS   | 11026B        |                                 |
|        | ENDM  |               |                                 |
| DRT    | MACRO | D             |                                 |
|        | SKS   | 10026B        |                                 |
|        | ENDM  |               |                                 |
| ALR    | OPD   | 00610026B,2   | ALERT RAD                       |
| ALRS   | OPD   | 00610226B,2   | READ SECTOR ADDRESS             |
| RRF    | OPD   | 00602226B,2   | READ RAD FILE                   |
| WRF    | OPD   | 00602266B,2   | WRITE RAD FILE                  |
| IOSDW  | EQU   | 00216200B     | W CHAN IOSD                     |
| IOSDE  | EQU   | 00614200B     | E CHAN IOSD                     |
| IOSDEC | EQU   | 617200B       | RAD IOSD ARMING ZERO WORD COUNT |
| CETE   | MACRO |               |                                 |
|        | DATA  | 04051000B     |                                 |
|        | ENDM  | E             | CHAN ERROR TEST                 |

CATE MACRO; DATA 04054000B; ENDM  
 EOD OPD 00600000B,1,1  
 RDC MACRO; EOM 10226B; ENDM;\* RESET DISC CONTROLLER

\* PARAMETERS

|        |     |             |   |
|--------|-----|-------------|---|
| BIT0   | EQU | 40000000B   |   |
| BIT1   | EQU | 20000000B   |   |
| BIT2   | EQU | 10000000B   |   |
| BIT3   | EQU | 40000000B   |   |
| BIT4   | EQU | 20000000B   |   |
| BIT5   | EQU | 10000000B   |   |
| BIT6   | EQU | 4000000B    |   |
| BIT7   | EQU | 2000000B    |   |
| BIT8   | EQU | 1000000B    |   |
|        |     |             |   |
| MXUSRS | EQU | 1600        |   |
| RTCNT  | EQU | 64          | PAPER TAPE READER BUFFER LENGTH         |
| PNCNT  | EQU | 40          | PAPER TAPE PUNCH BUFFER LENGTH          |
| CRCNT  | EQU | 82          | CARD READER BUFFER LENGTH               |
| NTAPE  | EQU | 2           | NUMBER OF MAG TAPE UNITS                |
| NLINK  | EQU | 0           | NUMBER OF PDP-5 LINKS                   |
| TCNT   | EQU | 199         | LENGTH OF MAG TAPE BUFFER               |
| LPCNT  | EQU | 120         | LINE PRINTER BUFFER LENGTH              |
| RTWT   | EQU | RTCNT*40/3  | PAPER TAPE READ TIME                    |
| PNWT   | EQU | PNCNT*400/6 | PAPER TAPE PUNCH TIME                   |
| CRWT   | EQU | 300         | CARD READ TIME                          |
| TXWT   | EQU | 20+TCNT/10  | MAG TAPE TIME                           |
| LPWT   | EQU | 133         | LINE PRINT TIME                         |
| NTRTRY | EQU | 10          | NUMBER OF REREADS                       |
| NTWTRY | EQU | 3           | NUMBER OF REWRITES                      |
| NFILE  | EQU | 15          | NUMBER OF FILES                         |
| UMSZ   | EQU | 16          | INITIAL MACHINE SIZE                    |
| NLTTC  | EQU | 0           |   |
| NTTYC  | EQU | 64          | NUMBER OF CHARS IN TTY BUFFER           |
| TTYEWM | EQU | 6           | TTY EARLY WARNING (.6 SEC)              |
| TTYEEW | EQU | 10          | TTY EXTRA EARLY WARNING (1 SEC)         |
| NFQU   | EQU | 30          | FULL QUANTUM SIZE                       |
| AMB    | EQU | 40000B      | ACCEPT MESSAGE BIT                      |
| AIB    | EQU | 100000B     | ACCEPT INPUT BIT                        |
| SRB    | EQU | 4000000B    | 8-LEVEL INPUT BIT                       |
| SPB    | EQU | 20000000B   | 8-LEVEL OUTPUT BIT                      |
| LISB   | EQU | 4000000B    | LISTEN FOR INPUT BIT                    |
| NPAC   | EQU | 100         | NUMBER OF PACT SLOTS                    |
| NPPAR  | EQU | 12          | LENGTH OF PACT ENTRY                    |
| NTTY   | EQU | 32          | NUMBER OF TTYS                          |
| NTTB   | EQU | NTTY+NLINK  | TOTAL TTY BUFFERS                       |
| NJOB1  | EQU | NTTB        | NUMBER OF JOBS WITHOUT P.U.             |
| NJOB   | EQU | NJOB1+1     | NUMBER OF JOBS                          |
| NMEM   | EQU | 32          | NUMBER OF PAGES                         |
| NSMEM  | EQU | 7           | NUMBER OF PAGES USED BY SYSTEM          |
| NPOP   | EQU | 43B         | NUMBER OF SYSPOPS IN USE                |
| NSQU   | EQU | 9           | NUMBER OF CLOCK CYCLES IN SHORT QUANTUM |
| NPUQ   | EQU | 15          | NUMBER OF PUCT ENTRIES                  |
| NSEC   | EQU | 2           |   |
| L2NSEC | EQU | 1           |   |

|  |       |                       |  |
|--|-------|-----------------------|--|
| NDRAT                                      | EQU   | 6                     |  |
| NCMEM                                      | EQU   | 57B                   | COMMON PART OF USER MACHINE                              |
| NSMT                                       | EQU   | 57B                   | SIZE OF SMT  |
| NUMEM                                      | EQU   | 100B-NCMEM            | NUMBER OF PRIVATE USER PAGES                             |
| UDA  | EQU   | 1000                  | INITIAL DRUM ASSIGNMENT                                  |
| NRRQ                                       | EQU   | NMEM-NSMEM            |  |
| NDRQ                                       | EQU   | 5                     | NUMBER OF DISC I/O REQUESTS                              |
| NBUFX                                      | EQU   | 3                     | NUMBER OF BUFFERS IN THE EXEC BLOCK                      |
| NBUF                                       | EQU   | NBUFX                 | TOTAL NUMBER OF BUFFERS                                  |
| NDBW                                       | EQU   | 631B                  | LENGTH OF DRUM BUFFER                                    |
| NDDW                                       | EQU   | 255                   | LENGTH OF DATA BLOCK                                     |
| NDXW                                       | EQU   | NDBW-NDDW-9           | LENGTH OF INDEX BLOCK                                    |
| NRTRY                                      | EQU   | 5                     | NUMBER OF READ TRIES FOR RAD                             |
| NDTRY                                      | EQU   | 3                     | NUMBER OF READ TRIES FOR DRUM                            |
| NSBND                                      | EQU   | 18                    | NUMBER OF 16K BANDS RESERVED FOR SWAPPING                |
| OSBND                                      | EQU   | 20                    | NUMBER OF 16K BANDS RESERVED FOR SYSTEM                  |
| DMPBND                                     | EQU   | 22                    | BAND FOR AUTOMATIC DUMP                                  |
| DBTOP                                      | EQU   | 37777B-NBUFX*NDBW-5   |  |
| *  |       |                       | BEGINNING OF DRUM BUFFERS IN EXEC TS                     |
| MBUFA                                      | EQU   | 03700000B             | FBWRD FOR SHARED BLOCK                                   |
| MBUFX                                      | EQU   | 34000000B             | FBWRD FOR EXEC BLOCK                                     |
| DCTX                                       | EQU   | 10000000B             | CONDITIONAL TRANSMIT COMMAND                             |
| DCBR                                       | EQU   | 20000000B             | BRANCH COMMAND   |
| DCDI                                       | EQU   | 70000000B             | DISCONNECT AND INTERRUPT COMMAND                         |
| DBB  | EQU   | 00400000B             | PROTECTED FILE BUSY BIT                                  |
|  | IF    | -COM                  |  |
| SMIFIL                                     | EQU   | DBTOP+1               | SECONDARY MEMORY INPUT FILE                              |
| SMBA                                       | EQU   | DBTOP+2               | SECONDARY MEMORY BUFFER ADDRESS                          |
| SMDRN                                      | EQU   | DBTOP+3               | SECONDARY MEMORY BDN ADDRESS                             |
| SMOFIL                                     | EQU   | DBTOP                 | SECONDARY MEMORY OUTPUT FILE                             |
| FBWA                                       | EQU   | 300000B               | SHARED BLOCK FBWRD ADDRESS.                              |
|  | ENDIF |                       |  |
| FBWRD                                      | EQU   | DBTOP+4               | BUFFER AVAILABILTY BIT WORD                              |
| BXO  | EQU   | NDBW-NDXW             | INDEX BLOCK ORIGIN REL TO BUFF                           |
| BBP  | EQU   | BXO+NDXW-1            | BACKWARD CHAIN WORD                                      |
| BFP  | EQU   | BXO+NDXW-2            | FORWARD CHAIN WORD                                       |
| BIC  | EQU   | NDDW+4                | INDEX CHANGED FLAG                                       |
| BIP  | EQU   | NDDW+7                | INDEX BLOCK POINTER                                      |
| BDC  | EQU   | NDDW+6                | CHANGED DATA FLAG  |
| BIA  | EQU   | NDDW+8                | INDEX BLOCK DRUM ADDRESS                                 |
| BDN  | EQU   | NDDW+5                | DATA BLOCK NUMBER  |
| BIN  | EQU   | NDDW+3                | INDEX BLOCK NUMBER                                       |
| ***PARAMETERS FOR DSC AND MWS0. 6/1/68. DC |       |                       |  |
| NACCTS                                     | EQU   | 400B                  | NUMBER OF USER ACCOUNTS                                  |
| LBDRL                                      | EQU   | 4                     | LIBRARY DIRECTORY LENGTH                                 |
| NDISCS                                     | EQU   | 20                    | NUMBER OF AVAILABLE DISCS                                |
| NBLKS                                      | EQU   | NDISCS*4000B          |  |
| NSECTS                                     | EQU   | NBLKS*4               |  |
| FAVDS                                      | EQU   | NACCTS+4*MXUSRS+LBDRL | FIRST AVAILABLE DISC SECTOR                              |
| NAVDS                                      | EQU   | NSECTS-FAVDS          | NUMBER OF AVAILABLE DISC SECTORS                         |
| NAVDB                                      | EQU   | NAVDS/4               | NUMBER OF AVAILABLE BLOCKS                               |
| TOP  | EQU   | 56000B                | THE DISC ADDRESSES FAVDS,FAVDS+4...TOP ARE IN THE WINDOW |
| NWINBLKS                                   | EQU   | TOP/4-FAVDS/4         | NUMBER OF BLOCKS REPRESENTED IN THE WINDOW               |
| WINLEN                                     | EQU   | NWINBLKS/24           | WINDOW LENGTH  |

NFWD EQU NAVDB/24 NUMBER OF FULL WORDS IN THE BITMAP  
 REM EQU NAVDB-NFWD\*24  
 SHIFTER EQU 24-REM TO ESTABLISH THE LAST WORD OF THE BITMAP  
 IF REM;LTAB EQU NFWD+1; ELSE;LTAB EQU NFWD; ENDF LENGTH OF BITMAP  
 \*\*\*END OF ADDITIONS MADE 6/1/68

CLKTIM EQU 60 CLOCK TIME

\* MACROS

|               |       |  |
|---------------|-------|--|
|               | IF    | ARMF   |
| ARMI          | MACRO | D  |
|               | AIR   |  |
|               | POT   | D(1)   |
|               | ENDM  |  |
|               | ELSF  | 1  |
| ARMI          | MACRO |  |
|               | ENDM  |  |
|               | ENDF  |  |
| ENTRY         | MACRO | L  |
| ENTCNT        | NARG  |  |
|               | RPT   | ENTCNT   |
| L(ENTCNT) EXT |       |  |
| ENTCNT        | EQU   | ENTCNT-1                                       |
|               | ENDR  |  |
|               | ENDM  |  |
| SETINT        | MACRO | A  |
|               | LDA   | =A(1)  |
|               | STA   | BLK31  |
|               | ENDM  |  |
| TDT           | MACRO | L  |
| L(1).W        | EQU   | *  |
|               | RPT   | NTAPE  |
|               | L(2)  | L(3).B+*-L(1).W                                |
|               | ENDR  |  |
|               | ENDM  |  |
| RMFF          | MACRO |  |
|               | ENDM  |  |
| SMFF          | MACRO |  |
|               | ENDM  |  |
| READ          | MACRO | D,G,1  |
| G(1)          | RSR   |  |
|               | BRU   | *-1  |
|               | ALR   |  |
|               | POT   | =D(3)/100B                                     |
|               | EOD*  | 10000B   |
|               | DATA  | IOSDE+D(1)/2000B(AND)37B+D(2)/40000B(AND)3*40B |
|               | POT   | =D(1)(AND)1777B*40000B+D(2)(AND)37777B         |
|               | RRF   |  |
|               | RSR   |  |
|               | BRU   | *-1  |
|               | RSE   |  |
|               | BRU   | G(1)   |
|               | CETE  |  |
|               | BRU   | G(1)   |
|               | ENDM  |  |
| ECHR          | MACRO | N  |

```

ECHRWD EQU ECHRWD*400B+N(1).B
NECHR EQU NECHR+1
      IF NECHR-2
      DATA ECHRWD
ECHRWD EQU 0
NECHR EQU 0
      ENDF
      ENDM

ECHB MACRO N
ECHVB EQU N(1)B
      RPT N(2)
      ECHR ECHV
ECHVB EQU ECHVB+N(3)
      ENDR
      ENDM

TRP MACRO L
ENTCNT NARG ENT CNT
      RPT ENT CNT
L(ENTCNT) EQU TRAP
      FRGT L(ENTCNT)
ENTCNT EQU ENT CNT-1
      ENDR
      ENDM

CACR MACRO D
      IF D(1)
      BRU CACCS
      BRU CACCF
      ELSF 1
      BRU CACCF
      BRU CACCS
      ENDF
      ENDM

LBL MACRO D
1LBL EQU D(2)
      RPT D(2)
      LDA D(1)+1LBL-1
      LRSH 6
1LBL EQU 1LBL-1
      ENDR
      ENDM

EXECI EQU 10000B
EXECF EQU 10001B
EXECT EQU 10002B
EXECR EQU 10003B

DB MACRO D
ENTCNT EQU D(1)*400B+D(2)*100B
++400000000B+D(4)(AND)1*40B+D(4)(AND)2*40000B
      RPT D(3)
      DATA ENT CNT
ENTCNT EQU ENT CNT+100B
      ENDR

```

ENDM

FORGT MACRO D  
ENTCNT NARG  
RPT ENTCNT  
FRGT D(ENTCNT)  
ENTCNT EQU ENTCNT-1  
ENDR  
ENDM

FORGT CRXF,AIB,8PB,8RB,AMB,LISB  
FORGT MBUFA,MBUF X,PNXF,LPXF  
FORGT RTCNT,PNCNT,CRCNT,TCNT,LPCNT  
FORGT RTWT,PNWT,CRWT,TXWT,LPWT  
FORGT NTAPE,NLINK,NBUF X,NBUF  
FORGT NTRTRY,NTWTRY,NDTRY  
FORGT UDA,NDDW,NDXW,NDBW,DBTOP  
FORGT NFILE,UMSZ,NTTYC,TTYEWM  
FORGT NPAC,NPPAR,NJOB,NJOB1,NFQU,NSQU  
FORGT NPOP,NMEM,NSMEM,NCMEM,NUMEM  
FORGT NPUQ,NSEC,L2NSEC,NDRQ,NSBND,NSAM,NSMT  
FORGT DCTX,DCBR,DCDI,DBB,ENTCNT  
FORGT FBWRD  
FORGT BXO,BBP,BFP,BIN,BIC,BDN,BDC,BIP,BIA  
IF -COM  
FORGT SMIFIL,SMBA,SMDRN,SMOFIL  
ENDF

FREEZE

ENTRY GMT,DMT,DPUCT,DFCB,DJOB,DIPMT,DAPMT,DDRQ  
ENTRY DRMT,DPACT,DQST,DTTY

\*  
\* DECODE MONITOR TABLES AND PRINT  
\*

\*  
\* UNPACK TABLE -- BRM UNPACK FOLLOWED BY ITEM LIST THEN -1.  
\* EACH ITEM IS A 5-BIT LEFT BIT NUMBER FOLLOWED BY A 5-BIT  
\* RIGHT SHIFT COUNT AND A 14-BIT ADDRESS INCREMENT.  
\* THE UNPACKED WORDS ARE STORED IN THE UT TABLE: THE FIRST  
\* TWO WORDS ARE THE ADDRESS OF THE CALL TO UNPACK AND THE  
\* CONTENTS OF THE X REGISTER (I.E. THE TABLE BASE) ON ENTRY.  
\*

IF DEBUG  
US1 ZR0  
US2 ZR0  
UT BSS 100  
UNPACK ZR0  
STX US1  
STX UT+1  
LDA =UT+2  
STA US2

```

UP1   LDA    UNPACK
      STA    UT
      MIN    UNPACK
      LDA*   UNPACK
      SKE    =-1
      BRU    ++2
      BRR    UNPACK
      ADD    US1
      CAX
      LDA*   UNPACK
      LRSR   19
      LDX    0,2
      XXA
      LCV    0,2
      STA*   US2
      LDA*   UNPACK
      RCV    14
      ETR    =37B
      CAX
      LDA*   US2
      LRSR   0,2
      STA*   US2
      MIN    US2
      BRU    UP1

```

```

*
* THE ITEM MACRO TAKES 1 ARGUMENT (DENOTING A FULL WORD),
* 2 ARGUMENTS (DENOTING A SINGLE BIT), OR 3 ARGUMENTS (GIVING
* THE FIRST AND LAST BIT NUMBERS). THE ILIST MACRO TAKES A
* LIST OF ARGUMENTS EACH OF WHICH IS AN ARGUMENT LIST FOR ITEM.
*

```

```

ITEM  MACRO  D
1I    NARG
      IF     1I=1
      ITEM  D(1),0,23
      ELSF   1I=2
      ITEM  D(1),D(2),D(2)
      ELSF   1
1I    EQU   23-D(3)+D(2)
2I    EQU   D(2)(AND)3*16+1I/2
      IF     1I(AND)1
      ($2I)* D(1),D(2)/4
      ELSF   1
      ($2I)  D(1),D(2)/4
      ENDF
      ENDF
      ENDM
      FRGT   1I
      FRGT   2I
ILIST MACRO  L
1L    NARG
2L    EQU   1
      RPT   1L
      ITEM  L(2L)

```

```

2L      EQU      2L+1
        ENDR
        ENDM
        FRGT     1L
        FRGT     2L

```

```

*
* THE STEP AND SEQ MACROS MAY BEST BE EXPLAINED IN QUASI-ALGOL:
* STEP(VAR,INIT,INCR,LIMIT,PROC)=FOR VAR=INIT,
* VAR+INCR WHILE VAR#LIMIT DO PROC(VAR)

```

```

* SEQ(VAR,INIT,LIMIT,PROC)=FOR VAR=INIT,VAR WHILE VAR#LIMIT
* DO VAR=PROC(VAR)

```

```

*
STEP    MACRO    L
        LDA      L(2)
        STA      L(1)
        CAX
        BRM      L(5)
        LDA      L(1)
        ADD      L(3)
        SKE      L(4)
        BRU      *-6
        ENDM

```

```

SEQ     MACRO    L
        LDA      L(2)
        BRU      **4
        STA      L(1)
        CAX
        BRM      L(4)
        SKE      L(3)
        BRU      *-4
        ENDM

```

```

*
* DECODE FORMAT STRING -- BRM DECODE FOLLOWED BY THE FORMAT
* AS AN ASCII STRING. UNRECOGNIZED CHARACTERS IN THE FORMAT
* ARE TYPED OUT. THE FOLLOWING CHARACTERS HAVE SPECIAL MEANINGS:
* SLASH - CARRIAGE RETURN AND LINE FEED
* DOT - END OF FORMAT STRING
* QUOTE - TYPE LITERALLY UNTIL NEXT QUOTE
* THE FOLLOWING CHARACTERS ARE FOLLOWED BY A 2-DIGIT DECIMAL
* INTEGER WHICH IS AN INDEX IN UT:
* PLUS - PRINT AS UNSIGNED OCTAL
* MINUS - PRINT AS SIGNED OCTAL (SIGN EXTENDED IF NECESSARY)
* LPAR, UT INDEX, ANYTHING, RPAR - PROCESS IF NONZERO
* EQUALS - REVERSE AND PRINT AS BINARY

```

```

DS1     DATA    0,137777B
DS2     ZRO
DECODE  ZRO
        LDA      DECODE
        ETR      =37777B
        MUL      =3
        LSH      23

```

|      |     |            |         |
|------|-----|------------|---------|
|      | ADD | =2         |         |
|      | STA | DS1        |         |
| DC1  | GCI | DS1        |         |
|      | HLT |            |         |
|      | SKE | =16B       | (DOT)   |
|      | BRU | DC2        |         |
|      | LDA | DS1        |         |
|      | MUL | =12525253B |         |
|      | STA | DECODE     |         |
|      | BRR | DECODE     |         |
| DC2  | SKE | =2         | (QUOTE) |
|      | BRU | DC13       |         |
| DC4  | GCI | DS1        |         |
|      | HLT |            |         |
|      | SKE | =2         |         |
|      | BRU | *+2        |         |
|      | BRU | DC1        |         |
|      | CIO | =1         |         |
|      | BRU | DC4        |         |
| DC13 | SKE | =17B       | (SLASH) |
|      | BRU | DC3        |         |
|      | TCO | =155B      |         |
|      | TCO | =152B      |         |
|      | BRU | DC1        |         |
| DC3  | SKE | =10B       | (LPAR)  |
|      | BRU | DC6        |         |
|      | BRM | DAX        |         |
|      | SKE | =0         |         |
|      | BRU | DC1        |         |
|      | STA | DS2        |         |
| DC7  | GCI | DS1        |         |
|      | HLT |            |         |
|      | SKE | =10B       |         |
|      | BRU | DC5        |         |
|      | MIN | DS2        |         |
|      | BRU | DC7        |         |
| DC5  | SKE | =11B       |         |
|      | BRU | DC7        |         |
|      | SKR | DS2        |         |
|      | BRU | DC7        |         |
|      | BRU | DC1        |         |
| DC6  | SKE | =13B       | (PLUS)  |
|      | BRU | DC8        |         |
|      | BRM | DAX        |         |
| DC9  | LDB | =8         |         |
|      | LDX | =1         |         |
|      | BRS | 36         |         |
|      | BRU | DC1        |         |
| DC8  | SKE | =15B       | (MINUS) |
|      | BRU | DC14       |         |
|      | BRM | DAX        |         |
|      | LDX | DB2        |         |
|      | LCY | 0,2        |         |
|      | RSH | 0,2        |         |

|      |      |       |          |
|------|------|-------|----------|
|      | SKG  | =-1   |          |
|      | BRU  | *+2   |          |
|      | BRU  | DC9   |          |
|      | TCO  | =15B  |          |
|      | CNA  |       |          |
|      | BRU  | DC9   |          |
| DC14 | SKE  | =35B  | (EQUALS) |
|      | BRU  | DC12  |          |
|      | BRM  | DAX   |          |
|      | STA  | DS2   |          |
|      | LDA  | =23   |          |
|      | SUB  | DB2   |          |
|      | STA  | DB2   |          |
| DC15 | LDA  | DS2   |          |
|      | RCY  | 1     |          |
|      | STA  | DS2   |          |
|      | LDA  | =10B  |          |
|      | LCY  | 1     |          |
|      | CIO  | =1    |          |
|      | SKR  | DB2   |          |
|      | BRU  | DC15  |          |
|      | BRU  | DC1   |          |
| DC12 | SKE  | =11B  | (RPAR)   |
|      | CIO  | =1    |          |
|      | BRU  | DC1   |          |
| *    |      |       |          |
| DB1  | ZRO  |       |          |
| DB2  | ZRO  |       |          |
| DAX  | ZRO  |       |          |
|      | GCI  | DS1   |          |
|      | HLT  |       |          |
|      | MUL  | =5    |          |
|      | STB  | DS2   |          |
|      | GCI  | DS1   |          |
|      | HLT  |       |          |
|      | ADD  | DS2   |          |
|      | SUB  | =260B |          |
|      | CAX  |       |          |
|      | ADD  | UT    |          |
|      | SUB  | =1    |          |
|      | STA  | DB2   |          |
|      | LDA* | DB2   |          |
|      | LRSH | 19    |          |
|      | STA  | DB1   |          |
|      | CLA  |       |          |
|      | LCY  | 5     |          |
|      | STA  | DB2   |          |
|      | LDA  | UT,2  |          |
|      | BRR  | DAX   |          |

\*  
\* DECODING ROUTINES FOR MONITOR TABLES

\*

\*

\* GET MONITOR TABLES TO USER CORE

\*

GMT ZRO; LDX =-30000B; STA 30000B,2; EAX 3777B,2; BRX \*-2  
BRS 57; BRS 7; BRR GMT

\*

\* DECODE ALL MONITOR TABLES

\*

DMT ZRO  
BRM BRM DPUCT  
BRM BRM DFCE  
BRM BRM DJOB  
BRM BRM DDRQ  
BRM BRM DQST  
BRM BRM DAPMT  
BRM BRM DSMT  
BRM BRM DRMT  
BRM BRM DTTY  
BRM BRM DPACT  
BRR DMT

IF -1

\*

\* DECODE SWAPPER ASSOCIATIVE MEMORY

\*

DSAMN ZRO  
DSAM ZRO  
BRM DECODE  
ASC '/\*\*\* SAM \*\*\*/.'  
SKN SACNT  
BRR DSAM  
STEP DSAMN, SACNT, =1, =0, DSAMI  
BRR DSAM

DSAMI ZRO  
BRM UNPACK  
ILIST (SAM1,0,5),(SAM1,6,11),(SAM1,12,17)  
ILIST (SAM1,18,23),(SAM2,0,5),(SAM2,6,11),(SAM2,12,17)  
ILIST (SAM2,18,23),(SAM3,0,11),(SAR1,0,5),(SAR1,6,11)  
ILIST (SAR1,12,17),(SAR1,18,23),(SAR2,0,5),(SAR2,6,11)  
ILIST (SAR2,12,17),(SAR2,18,23),(SAR3,12,17),(SAR3,18,23)  
ILIST (SAM3,12,17),(SAM3,18,23)  
DATA -1  
BRM DECODE  
ASC 'JOB +10: +02,+03,+04,+05 +06,+07,+08,+09 +21,+22/  
+ +11,+12,+13,+14 +15,+16,+17,+18 +19,+20/.'  
BRR DSAMI  
ENDF

\*

\* DECODE PHANTOM USER QUEUE

-----

```

*
DPUCTN ZRO
DPUCT  ZRO
      LDA  PUCTR
      STA  UT+1
      BRM  DECODE
      ASC  '/*** PUCT ***/PUCTR "=" +01//.'
      SEQ  DPUCTN,PUBPTR,=PUBPTR,DPUCTI
      BRR  DPUCT
DPUCTI ZRO
      BRM  UNPACK
      ILIST 0,(1,0,8),(1,9,23),(2,0,11),(2,12,23)
      DATA -1
      BRM  DECODE
      ASC  'TEST +03"/"+04 PUPAC -05 FILE +06//.'
      LDA  UT+2
      BRR  DPUCTI

```

```

*
* DECODE FILE CONTROL BLOCKS
*

```

```

DFCBN ZRO
DFCB  ZRO
      BRM  DECODE
      ASC  '/*** FCBS ***/.'
      STEP DFCBN,=3,=1,=NFILE,DFCBI
      BRR  DFCB
DFCBI ZRO
      LDA  FA,2
      SKA  =40000000B
      BRR  DFCBI
      BRM  UNPACK
      ILIST (FA,0,6),(FD,9,23),(FC,3,8),(FC,9,23),(FA,9,23)
      ILIST (FC,0,2),(FD,2),(FD,8),(FD,1),(FD,6)
      ILIST (FD,3),(FD,4),(FD,5),(FD,0),FW
      DATA -1
      BRM  DECODE
      ASC  'FILE +01 JOB +02 DEVICE +03(04 UNIT +04)
+(05 BUFFER +05)/ (06ADDR +06 )COUNT -07(08 CH)(09 OUT)(10 BUSY)
+(15 ERR)(11 PUB)(12 DRM)(13 RX)(14 RO)(16 WRD +16)//.'
      BRR  DFCBI

```

```

*
* DECODE JOB INDEXED TABLES
*

```

```

DJ0BN ZRO
DJOB  ZRO
      BRM  DECODE
      ASC  '/*** JOBS ***/.'
      STEP DJ0BN,=1,=1,=NJOB,DJOBI
      BRR  DJOB
DJOBI ZRO
      LDA  PMTP,2
      SKG  =0
      BRR  DJOBI
      BRM  UNPACK

```

```

        ILIST      (RL3,12,17),(RL3,18,23)
        ILIST      (PMA,1),(PMA,3,8),(PMA,12,17),(PMA,18,23),TTNO
        DATA      -1
        BRM         DECODE
        ASC         'JOB +01 TTY +08 RL3 +02,+03.'
+ NPMT +07 USED +06 UNUSED +05 (04 NO CHARGE)/.
        LDX         UT+8
        LDA         TTYASG,2
DJOB1  CAX
        LDA         PPTR,2
        MRG         PLMSK
        SKA         PRMSK
        BRU         DJOB1
        LDA         =DJOBP
        BRM         MFS
        BRM         DECODE
        ASC         '/.'
        BRR         DJOB1
DJOBP  ZRO
        CAX
        BRU*        **2,2
        DATA      DJPU,DJPP,DJPD
DJPU   BRM         DECODE
        ASC         '""'.
        BRR         DJOBP
DJPP   BRM         DECODE
        ASC         ',-01.'
        BRR         DJOBP
DJPD   BRM         DECODE
        ASC         '""(-01.'
        BRR         DJOBP

```

```

*
* DECODE RAD BIT TABLE
*

```

```

        IF -1
DDBTN  ZRO
DDBT   ZRO
        BRM         DECODE
        ASC         '/*** DRAT ***/.'
        STEP        DDBTN,=0,=1,=NSEC,DDRATI
        BRR         DDBT
DDRATI ZRO
        BRM         UNPACK
        ILIST      (DRAT1,0,4),(DRAT2,0,23),(DRAT3,8,23)
        DATA      -1
        BRM         DECODE
        ASC         'SECTOR +01: =02 =03/.'
        BRR         DDRATI
        ENDF

```

```

*
* DECODE SMT
*

```

```

DSMTN  ZRO
DSMT   ZRO

```

```

BRM      DECODE
ASC      '/*** SMT ***//.'
STEP     DSMTN,=0,=1,=NSMT,DSMTI
BRR      DSMT
DSMTI    ZRO
LDA      SMT,2
SKA      =77700B
SKA      X4
BRR      DSMTI
BRM      UNPACK
ILIST    (SMT,2,7),(SMT,8),(SMT,9,15),(SMT,16,17)
ILIST    (SMT,18),(SMT,19,20)
DATA     -1
BRM      DECODE
ASC      '+01 +04".'+05 +07(02 UC +02)(03 EX)(06 RO)/.'
BRR      DSMTI

```

```

*
* DECODE INDIVIDUAL PMT
*

```

```

DIPMTN  ZRO
DIPMTL  ZRO
DIPMTU  ZRO
DIPMT   ZRO
LDA     PMTP,2
STA     DIPMTL
STA     DIPMTU
SKG     =0
BRR     DIPMT
LDA     PMA,2
ETR     =77B
ADM     DIPMTU
STX     UT+1
BRM     DECODE
ASC     '/* PMT +01 *//.'
STEP    DIPMTN,DIPMTL,=1,DIPMTU,DPMTI
BRR     DIPMT
DPMTI   ZRO
LDA     0,2
SKE     =0
BRU     *+2
BRR     DPMTI
BRM     UNPACK
ILIST   (0,0),(0,1),(0,8),(0,9,15),(0,16,17),(0,18),(0,19,23)
DATA    -1
LDA     DIPMTL
CNA
ADD     =NCMEM
ADM     UT+1
BRM     DECODE
ASC     '+01 .'
LDA     UT+3
SKE     =0
BRU     DPMTS
BRM     DECODE

```

```

ASC      '+05"'+06 .'
LDA      UT+2
SKE      =0
BRU      DPMTI1
BRM      DECODE
ASC      '+08.'
BRU      DPMTI2
DPMTI1  BRM      DECODE
ASC      'DRM.'
DPMTI2  BRM      DECODE
ASC      '(04 EX)(07 RO)/'.
BRR      DPMTI
DPMTS   LDX      US1
BRM      UNPACK
ILIST    (0,9,23)
DATA     -1
BRM      DECODE
ASC      '"("SMT +02")"'.
BRR      DPMTI

```

```

*
* DECODE ALL PMTS

```

```

*
DAPMTN  ZRO
DAPMT   ZRO
BRM      DECODE
ASC      '/* ** PMTS **/'.
STEP    DAPMTN,=1,=1,=NJOB,DIPMT
BRR      DAPMT

```

```

*
* DECODE DRUM COMMAND QUEUE

```

```

*
IF -1
DRRQN   ZRO
DRRQ    ZRO
BRM      DECODE
ASC      '/* ** RRQ **/'.
SEQ     DRRQN,ERCL,=0,DRRQI
BRR      DRRQ
DRRQI   ZRO
BRM      UNPACK
ILIST    (0,10,23),(1,3),(1,8,23)
ILIST    (2,4,10),(2,4,15),(3,12,17),(3,18,23),2,3
ILIST    (1,8,12),(1,13,23),(2,11,12)
DATA     -1
LDA      UT+2
SKE      =0
BRU      DRRQI2
DRRQI1  LDA      UT+1
ADD      =4
SKE      =RRQU
BRU      *+2
LDA      =RRQ
SKE      ERCL
BRR      DRRQI

```

|        |       |  |
|--------|-------|--|
|        | CLA   |  |
|        | BRR   | DRRQI  |
| DRRQI2 | BRM   | DECODE   |
|        | ASC   | '(03W )..'   |
|        | LDA   | UT+3   |
|        | SKE   | =0   |
|        | BRU   | DRRQI3   |
|        | BRM   | DECODE   |
|        | ASC   | 'R .'  |
| DRRQI3 | LDA   | UT+2   |
|        | SKE   | =4000B   |
|        | BRU   | DRRQI4   |
|        | BRM   | DECODE   |
|        | ASC   | 'P +05".." +13.'                                       |
| DRRQI5 | BRM   | DECODE   |
|        | ASC   | ' +11".." +12(07, FILE +07)(08, IR +08)/.'             |
|        | BRU   | DRRQI1   |
| DRRQI4 | SKE   | =NDDW  |
|        | BRU   | DRRQI6   |
|        | BRM   | DECODE   |
|        | ASC   | 'D.'   |
|        | BRU   | DRRQI7   |
| DRRQI6 | SKE   | =NDXW  |
|        | BRU   | DRRQI8   |
|        | BRM   | DECODE   |
|        | ASC   | 'I.'   |
| DRRQI7 | BRM   | DECODE   |
|        | ASC   | ' +06.'  |
|        | BRU   | DRRQI5   |
| DRRQI8 | BRM   | DECODE   |
|        | ASC   | 'WR +02 CA +04 DA +09 DC +10/.'                        |
|        | BRU   | DRRQI1   |
|        | *     |  |
|        | ENDF  |  |
| DDRQN  | ZRO   |  |
| DDRQ   | ZRO   |  |
|        | BRM   | DECODE   |
|        | ASC   | '/** DRQ **//.'  |
|        | SEQ   | DDRQN, EDCL, =0, DDRQI                                 |
|        | BRR   | DDRQ   |
| DDRQI  | ZRO   |  |
|        | BRM   | UNPACK   |
|        | ILIST | (0, 10, 23), (1, 3), (1, 8, 23)                        |
|        | ILIST | (2, 4, 10), (2, 4, 15), (3, 12, 17), (3, 18, 23), 2, 3 |
|        | ILIST | (1, 8, 12), (1, 13, 23), (2, 11, 12)                   |
|        | DATA  | -1   |
|        | LDA   | UT+2   |
|        | SKE   | =0   |
|        | BRU   | DDRQI2   |
| DDRQI1 | LDA   | UT+1   |
|        | ADD   | =4   |
|        | SKE   | =DRQU  |
|        | BRU   | *+2  |
|        | LDA   | =DRQ   |

|        |     |   |
|--------|-----|---|
|        | SKE | EDCL                                      |
|        | BRR | DDRQ1                                     |
|        | CLA |   |
|        | BRR | DDRQ1                                     |
| DDRQ12 | BRM | DECODE                                    |
|        | ASC | '(03W )..'                                |
|        | LDA | UT+3                                      |
|        | SKE | =0  |
|        | BRU | DDRQ13                                    |
|        | BRM | DECODE                                    |
|        | ASC | 'R . '                                    |
| DDRQ13 | LDA | UT+2                                      |
|        | SKE | =4000B                                    |
|        | BRU | DDRQ14                                    |
|        | BRM | DECODE                                    |
|        | ASC | 'P +05".." +13.'                          |
| DDRQ15 | BRM | DECODE                                    |
|        | ASC | ' +11".." +12(07, FILE +07)(08, IR +08)/' |
|        | BRU | DDRQ11                                    |
| DDRQ14 | SKE | =NDDW                                     |
|        | BRU | DDRQ16                                    |
|        | BRM | DECODE                                    |
|        | ASC | 'D.'                                      |
|        | BRU | DDRQ17                                    |
| DDRQ16 | SKE | =NDXW                                     |
|        | BRU | DDRQ18                                    |
|        | BRM | DECODE                                    |
|        | ASC | 'I.'                                      |
| DDRQ17 | BRM | DECODE                                    |
|        | ASC | ' +06.'                                   |
|        | BRU | DDRQ15                                    |
| DDRQ18 | BRM | DECODE                                    |
|        | ASC | 'WR +02 CA +04 DA +09 DC +10/'            |
|        | BRU | DDRQ11                                    |

\*  
\* DECODE RMT

|       |            |                             |
|-------|------------|-----------------------------|
|       | *<br>DRMTN | ZRO                         |
|       | DRMT       | ZRO                         |
|       |            | BRM                         |
|       |            | DECODE                      |
|       |            | '/*** RMT ***/.'            |
|       | STEP       | DRMTN,=NSMEM,=1,=NMEM,DRMTI |
|       | BRR        | DRMT                        |
| DRMTI | ZRO        |                             |
|       | BRM        | UNPACK                      |
|       | ILIST      | (RMT,0),(RMT,1),(RMT,10,23) |
|       | ILIST      | RMC,RMA                     |
|       | DATA       | -1                          |
|       | BRM        | DECODE                      |
|       | ASC        | ' +01 . '                   |
|       | LDA        | UT+4                        |
|       | SKE        | =0                          |
|       | BRU        | DRMTI1                      |
|       | BRM        | DECODE                      |

```

ASC          *UNUSED.*
DRMTI2 BRM   DECODE
ASC          * -05 +06/.*
BRR         DRMTI
DRMTI1 ADD   =1
SKG         =PMT
BRU         DRMTI6
CLX
DRMTI3 EAX   1,2
SKG         PMTP,2
BRU         DRMTI3
STX         UT+7
LDA         PMA,2
ETR         =77B
ADD         PMTP,2
SKG         UT+4
BRU         DRMTI5
LDA         PMTP,2
CNA
ADD         =NCMEM
ADM         UT+4
BRM         DECODE
ASC         *+07,+04.*
DRMTI7 BRM   DECODE
ASC         *(03 R0).*
BRU         DRMTI2
DRMTI6 LDA   =SMT
CNA
ADM         UT+4
BRM         DECODE
ASC         * +04.*
BRU         DRMTI7
DRMTI5 LDA   UT+4
ADD         =1
BRU         DRMTI3

```

```

*
* UNPACK PACT ENTRY
*

```

```

UPPE  ZR0
      BRM   UNPACK
      ILIST PNEXT,(PL,10,23),PA,PB,PX,RL1,RL2
      ILIST (PTEST,3,8),(PTEST,10,23),(PPTR,0,11),(PPTR,12,23),(PL,02
      ILIST (PQU,12,23),(PTAB,10,23),(PIM,4,23),(PQU,2,8),(PQU,9,11)
      ILIST (PL,3,8),(PQU,0),(PQU,1),(PTAB,0),(PIM,0),(PIM,1),(PIM,2)
      DATA -1
      BRR   UPPE

```

```

*
* MAP FORK STRUCTURE WITH (A)
*

```

```

MFSS  ZR0
MFT   BSS   10
MFP   ZR0
MFS   ZR0
      STA   MFSS

```

|      |      |       |
|------|------|-------|
|      | LDA  | ==1   |
|      | STA  | MFP   |
| MFS1 | BRM  | UPPE  |
|      | LDA  | =1    |
|      | MIN  | MFP   |
|      | LDX  | MFP   |
|      | STA  | MFT,2 |
|      | LDX  | UT+1  |
|      | BRM* | MFSS  |
| MFS4 | LDA  | UT+11 |
|      | SKE  | =0    |
|      | BRU  | MFS2  |
| MFS5 | LDA  | UT+14 |
|      | SKE  | =0    |
|      | BRU  | MFS3  |
|      | LDA  | ==1   |
|      | BRM* | MFSS  |
|      | SKR  | MFP   |
|      | BRU  | *+2   |
|      | BRR  | MFS   |
|      | LDA  | UT+12 |
|      | MRG  | PLMSK |
|      | CAX  |       |
|      | BRM  | UPPE- |

BRU MFSS

|      |      |       |
|------|------|-------|
| MFS3 | MRG  | PLMSK |
|      | LDX  | MFP   |
|      | MIN  | MFT,2 |
|      | CAX  |       |
|      | BRM  | UPPE  |
|      | CLA  |       |
|      | BRM* | MFSS  |
|      | BRU  | MFS4  |
| MFS2 | MRG  | PLMSK |
|      | CAX  |       |
|      | BRU  | MFS1  |

\*  
\* PRINT FORK STRUCTURE

|      |      |       |
|------|------|-------|
|      | PFSN | ZRO   |
|      | PFS  | ZRO   |
|      |      | CLX   |
| PFS1 | STX  | PFSN  |
|      | LDA  | MFT,2 |
|      | LDX  | =1    |
|      | LDB  | =8    |
|      | BRS  | 36    |
|      | LDA  | PFSN  |
|      | SKE  | MFP   |
|      | BRU  | PFS2  |
|      | BRR  | PFS   |
| PFS2 | ADD  | =1    |
|      | CAX  |       |
|      | TCO  | =16B  |

```

BRU      PFS1
SDIPACT ZRO; STX DPACTN; BRM DPACTI; BRR DIPACT
*
* DECODE ACTIVE PROCESS TABLE
*
DPACTN ZRO
DPACT  ZRO
      LDX      PACPTR
      STX      UT+1
      BRM      DECODE
      ASC      '/*** PACT ***/PACPTR "=" -01//.'
      STEP     DPACTN,=0,=1,=NJOB,DPACTI
      BRR      DPACT
DPACTI ZRO
      LDA      PMTP,2
      SKG      =0
      BRR      DPACTI
      STX      UT+1
      BRM      DECODE
      ASC      '/* JOB +01 */.'
      LDX      DPACTN
      LDX      TTNO,2
      LDA      TTYASG,2
DPACTI CAX
      LDA      PPTR,2
      MRG      PLMSK
      SKA      PRMSK
      BRU      DPACTI
      LDA      =DPACTE
      BRM      MFS
      BRR      DPACTI
DPACTE ZRO
      SKG      =-1
      BRR      DPACTE
      BRM      DECODE
      ASC      '/-01: .'
      BRM      PFS
      BRM      DECODE
      ASC      ' LOC +03(13,+13)(20 EX) TEST +09"/"+10/
+ A +04 B +05 X +06/ RL1 +07 RL2 +08 Q +17"("+18)"
+(21 EB)(22 LM)(23 FM)(24 TW)(25 NT)(16 IM +16)/.'
      BRR      DPACTE

```

```

*
* DECODE QUEUE STRUCTURE
*
DQSTS  ZRO
      ASC      ': /'
DQSTD  ZRO
      ASC      '$/'
DQSTA  ASC      'QTIQIQSQSQSQEQTI'
DQSTP  DATA    QTI,QIO,QSQ,QQE
DQSTN  ZRO
DQSTJ  ZRO
DQST   ZRO

```

```

BRM          DECODE
ASC          '/*** QUEUES ***//.'
STEP        DQSTN,=0,=1,=DQSTN-DQSTP,DQSTI
BRR         DQST
DQSTI       ZRO
BRM          UNPACK
ILIST       DQSTA,DQSTA+1,DQSTP
DATA        -1
LDP         UT+2
STA         DQSTS
STB         DQSTD
LDA*        UT+4
STA         DQSTJ
LDA         =DQSTS
LDB         =-1
LDX         =1
BRS         34
DQST5       LDA =8; STA DQST4
DQST1       LDA         DQSTJ
STA         UT+2
SKG         =-1
BRU         DQST2
LDX         UT+4
SKE         2,2
BRU         DQST3
LDA         =DQSTD
LDB         =-1
LDX         =1
BRS         34
BRR         DQSTI
DQST3       BRM          DECODE
ASC          '+02?//.'
BRR         DQSTI
DQST2       BRM          DECODE
ASC          '-02, .'
LDX         DQSTJ
LDA         PNEXT,2
STA         DQSTJ
SKR         DQST4
BRU         DQST1
TCO =155B; TCO =152B; BRU DQST5
DQST4       ZRO

```

```

*
* DECODE TELETYPE TABLES

```

```

*
DTTYN       ZRO
DTTY        ZRO
BRM          DECODE
ASC          '/*** TTYS ***//.'
STEP        DTTYN,=0,=1,=NTTB-1,DTTYI
BRR         DTTY
DTTYI       ZRO
BRM          UNPACK
ILIST       TTYASG

```

```
DATA      -1      (IMPROVE LATER)
LDA      ADMSK
SKG      UT+2
BRR      DTTYI
BRM      DECODE
ASC      '+01 -02/.'
BRR      DTTYI

ENDF
END
```