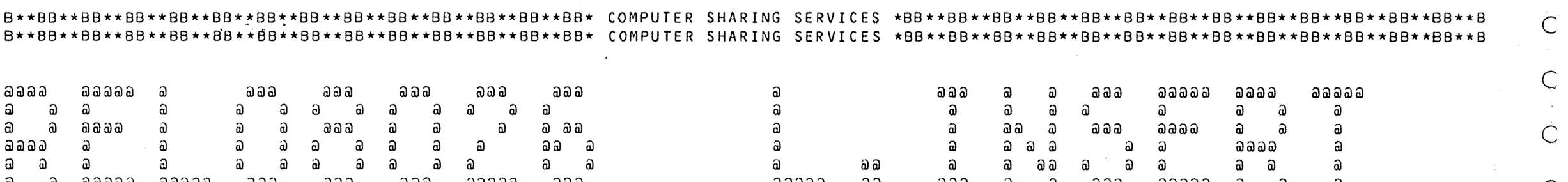
B**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB* COMPUTER SHARING SERVICES *BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**B
B**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB* COMPUTER SHARING SERVICES *BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**B



B**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB* COMPUTER SHARING SERVICES *BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**B
241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241

09/24/81 14:12:41 PRINTOUT #241

EXEC FOR BENCHMARKS
X SEGMENT

```
DDDD    AAA   V   V  EEEEE
D   D  A   A  V   V  E
D   D  AAAAA  V   V  E
D   D  A   A   V V   E
DDDD   A   A    V    EEEEE     ROBINSON
```

```
                INDEX
                TTL     DTSS EXECUTIVE (INSERT SEGMENT)                    DTSS TRADE SECRET
*
*
*       ****************************************************************
*       ****************************************************************
*       **                                                            **
*       **                                                            **
*       **           PROPRIETARY TRADE SECRET INFORMATION             **
*       **                                                            **
*       **  TO BE USED ONLY UNDER LICENSE FROM DTSS INCORPORATED.     **
*       **                                                            **
*       **                                                            **
*       **    UNPUBLISHED COPYRIGHTED WORK OF DTSS INCORPORATED.      **
*       **                                                            **
*       **                                                            **
*       ****************************************************************
*       ****************************************************************
*
*
*
*
                NAME    INSERT
*
```

```
                        THINGS STILL TO BE DONE

25              TTLS    THINGS STILL TO BE DONE
26      *
27      *       STORAGE MANAGEMENT
28      *            NEW ALLOCATION SYSTEM WITH BIT MAPS
29      *            DEALLOCATE SCRATCHED SCRATCH FILES
30      *            CHECK FOR POSSIBLE DEALLOCATION ON CLOSE OF CATALOGED FILES
31      *       STATISTICS
32      *            TOTAL USE OF SELECTED FILES
33      *       PHYSICAL I/O
34      *            RESET STATUS ON DEVICE DRIVES
35      *               FIX KLUDGE IN PIO WHERE TAPES AND CONSOLE USE SAME ERROR RECOVERY CODE.
36      *                 WHEN FIXED, THE DRIVE MME CALL TO PIO CAN USE A SHORTER LIST ELEMENT
37      *       MME CHANGES AND IMPROVEMENTS
38      *       MISCELLANEOUS
39      *            BETTER ERROR RECOVERY IN CATALOG ROUTINES
40      *            BETTER ERROR RECOVERY ON PARITY FAULTS
41      *            RECOVER FROM PAGE TABLE I/O ERRORS                                  [22SEP78]
42      *            WON'T BOOT WITH DEVICE 77
43      *            CHANGE DA FORMAT FOR LARGER MASS STORAGE DEVICES, MORE DEVICES      [30MAY80]
44      *            CHANGE CATALOG FORMAT FOR HASHED CATALOGS                           [09DEC79]
45      *            USE >1 CATALOG BUFFER                                               [09DEC79]
46      *                                                                                [09DEC79]
47      *                                                                                [09DEC79]
48      *       WHY IT'S NOT DONE:                                                       [09DEC79]
49      *                                                                                [09DEC79]
50      *       1)      WE DIDN'T THINK OF IT.                                           [09DEC79]
51      *                                                                                [09DEC79]
52      *       2)      WE COULDN'T FIND A GOOD WAY TO DO IT.                            [09DEC79]
53      *                                                                                [09DEC79]
54      *       3)      WE COULDN'T AGREE ON HOW TO DO IT.                               [09DEC79]
55      *                                                                                [09DEC79]
56      *       4)      IT WAS TOO RADICAL A CHANGE.                                     [09DEC79]
57      *                                                                                [09DEC79]
58      *       5)      IT WAS TOO EXPENSIVE TO IMPLEMENT.                               [09DEC79]
59      *                                                                                [09DEC79]
60      *       6)      IT WAS ONE MORE STRAW THAT WOULD BREAK THE CAMEL'S BACK.         [09DEC79]
61      *                                                                                [09DEC79]
62      *                               -S. J. GARLAND                                   [09DEC79]
```

```
                                DEFINITIONS -- IOM FLAG

                        63              TTLS    DEFINITIONS -- IOM FLAG
                        64      *
                        65      *        IOM VERSION, NEW SWAPPER
                        66      *
                000001  67      IOMFLG  EQU     1                                       [09DEC79]
                        68      *                                                       [09DEC79]
                        69      *       MACROS FOR USE W/                               [09DEC79]
                        70      *                                                       [09DEC79]
                        71      IFIOM   MACRO                                           [09DEC79]
                        72              IFE     IOMFLG,1,ENDIOM"ASSEMBLE IF IOM         [09DEC79]
                        73              ENDM    IFIOM                                   [09DEC79]
                        74      *                                                       [09DEC79]
                        75      IFIOC   MACRO                                           [09DEC79]
                        76              INE     IOMFLG,1,ENDIOC"SKIP IF IOM             [09DEC79]
                        77              ENDM    IFIOC                                   [09DEC79]
                        78      *                                                       [09DEC79]
                        79      *       NOTE: SOMETIMES THIS FLAG IS USED TO MEAN       [09DEC79]
                        80      *             6000 OR 66 INSTEAD OF 600.                [09DEC79]
```

DEFINITIONS --SYSTEM WIDE INTERESTING CONSTANTS

```
                    81              TTLSS   SYSTEM WIDE INTERESTING CONSTANTS                                [01SEP79]
                    82      *                                                                                [01SEP79]
                    83      *                                                                                [01SEP79]
                    84      *                                                                                [01SEP79]
                    85              HEAD    0                                                                [01SEP79]
                    86                                                                                       [01SEP79]
000002              87      NPROS   EQU     2               MAX NUMBER OF PROCESSORS ASSEMBLED FOR           [01SEP79]
                    88      *                                                                                [01SEP79]
                    89      *                                                                                [01SEP79]
                    90      *                                                                                [01SEP79]
000240              91      TOTJ    BOOL    240             MAX NUMBER OF SIMULTANEOUS JOBS ON SYSTEM        [01SEP79]
                    92      *                                                                                [30MAY80]
                    93      *                                                                                [30MAY80]
                    94      *       POINT AT WHICH EXTENDED MEMORY ADDRESSING IS INVOKED                     [30MAY80]
200000              95      FENCE   EQU     64*1024         32K WASN'T ENOUGH                                [30MAY80]
```

DEFINITIONS --LOW CORE LAYOUT

```
                    96              TTLSS   LOW CORE LAYOUT
                    97      *
                    98      *
                    99      *        IOM LOW CORE ALLOCATION
                   100      *
                   101              HEAD    V
                   102
          000000   103      INTV    BOOL    0               INTERRUPT VECTOR -- LENGTH = 100(8)
                   104
          000100   105      UTICK   BOOL    100             SPECIAL TIMEOUT TICKERS -- LENGTH = 100(8)      [22SEP78]
                   106                                                                                     [22SEP78]
          000200   107      USTAT   BOOL    200             DEVICE STATUS -- LENGTH 100(8)                  [22SEP78]
                   108                                                                                     [22SEP78]
          000300   109      USPEC   BOOL    300             SPECIAL INTERRUPT TASKS -- LENGTH = 100(8)      [22SEP78]
                   110                                                                                     [22SEP78]
          000400   111      PTYPE   BOOL    400             PHYSICAL DEVICE TYPE -- LENGTH = 100(8)         [22SEP78]
                   112                                                                                     [22SEP78]
          000500   113      UQS     BOOL    500             POINTERS TO DEVICE QUEUES -- LENGTH = 100(8)    [22SEP78]
                   114                                                                                     [22SEP78]
          000600   115      PDA     BOOL    600             PHYSICAL DEVICE ADDRESS -- LENGTH = 100(8)      [22SEP78]
                   116                                                                                     [22SEP78]
          000700   117      PREF    BOOL    700             PREFERENCE TABLE -- LENGTH = 100(8)             [22SEP78]
                   118                                                                                     [22SEP78]
          001000   119      PATCH   BOOL    1000            PATCH AREA -- LENGTH = 200(8)                   [22SEP78]
                   120                                                                                     [22SEP78]
          001200   121      ADA     BOOL    1200            PAGE TABLE DEVICE ADDRESSES -- LENGTH = 100(8)  [22SEP78]
                   122                                                                                     [22SEP78]
          001300   123      XSTAT   BOOL    1300            STATISTICS AREA POINTERS -- LENGTH = 40(8)      [22SEP78]
                   124                                                                                     [22SEP78]
          001340   125      IMW     BOOL    1340            INTERRUPT WORDS -- LENGTH = 40(8)               [22SEP78]
                   126                                                                                     [22SEP78]
          001400   127      MBX     BOOL    1400            MAILBOX AREA -- LENGTH = 200(8)                 [22SEP78]
                   128                                                                                     [22SEP78]
          001600   129      PTABS   BOOL    1600            FOUR INTERLEAVED TABLES OF STRIDE FOUR USED FOR [09DEC79]
                   130                                      PHYSICAL I/O -- LENGTH = 140(8)                 [22SEP78]
                   131                                                                                     [22SEP78]
          001740   132      PQS     BOOL    1740            QUEUES FOR CHANNELS -- LENGTH = 140(8)          [22SEP78]
                   133                                                                                     [22SEP78]
          002100   134      STTS    BOOL    2100            STATUS RETURN AREA -- LENGTH = 60(8)            [22SEP78]
                   135                                                                                     [22SEP78]
          002200   136      FTVS    BOOL    2200            FAULT VECTORS  -- LENGTH = 40(8) PER CP        [01SEP79]
                   137                                                                                     [01SEP79]
          002300   138      CFILE   EQU     32*$NPROS+FTVS  CORE FILE DA AREA -- LENGTH = 40(8)             [01SEP79]
                   139                                                                                     [01SEP79]
          002340   140      SISTK   EQU     32+CFILE        SYSTEM FAULT INTERRUPT STACK -- LENGTH = 40(8)  [01SEP79]
                   141
          003400   142      FVELS   BOOL    3400            DEFAULT FAULT VECTOR LOCATION ON DPS 8/4X ELS
```

V

DEFINITIONS --INDEX REGISTERS

```
                    143             TTLSS   INDEX REGISTERS
                    144     *
                    145     *
                    146     *       SYMBOLIC INDEX REGISTERS
                    147     *
                    148     *       THE COMMENTS DESCRIBE TYPICAL USAGE (E.G. IN MMES)                  [09DEC79]
                    149     *       SOME SEGMENTS FOLLOW OTHER CONVENTIONS (E.G. PIO)                   [09DEC79]
                    150     *
                    151             HEAD    0,A,C,E,H,I
000001              152     X       EQU     1               SCRATCH REGISTER
000002              153     Y       EQU     2               SCRATCH REGISTER
000003              154     Z       EQU     3               SCRATCH REGISTER
000004              155     T       EQU     4               POINTER TO LIST ELEMENT
000005              156     J       EQU     5               JOB NUMBER
000006              157     P       EQU     6               PERMANENT POINTER
000007              158     S       EQU     7               ABSOLUTE LOCATION OF CURRENT STATE VECTOR
                    159     *                                                                           [09DEC79]
000000              160     ...     EQU     0               SYMBOL TO INDICATE DATA TO BE FILLED IN     [09DEC79]
                    161     *
                    162             HEAD    K,Q,X,Z,L,N
000001              163     X       EQU     1               SCRATCH REGISTER
000002              164     Y       EQU     2               SCRATCH REGISTER
000003              165     Z       EQU     3               SCRATCH REGISTER
000004              166     T       EQU     4               POINTER TO LIST ELEMENT
000005              167     J       EQU     5               JOB NUMBER
000006              168     P       EQU     6               PERMANENT POINTER
000007              169     S       EQU     7               ABSOLUTE LOCATION OF CURRENT STATE VECTOR
                    170     *                                                                           [09DEC79]
000000              171     ...     EQU     0               SYMBOL TO INDICATE DATA TO BE FILLED IN     [09DEC79]
                    172     *
                    173             HEAD    W
000001              174     X       EQU     1               SCRATCH REGISTER
000002              175     Y       EQU     2               SCRATCH REGISTER
000003              176     Z       EQU     3               SCRATCH REGISTER
000004              177     T       EQU     4               POINTER TO LIST ELEMENT
000005              178     J       EQU     5               JOB NUMBER
000006              179     P       EQU     6               PERMANENT POINTER
000007              180     S       EQU     7               ABSOLUTE LOCATION OF CURRENT STATE VECTOR
                    181     *                                                                           [09DEC79]
000000              182     ...     EQU     0               SYMBOL TO INDICATE DATA TO BE FILLED IN     [09DEC79]
                    183     *
                    184     *       DEBUGGING SYMBOLS
                    185     *
                    186             HEAD    0
000011              187     DEBUG   EQU     9
                    188     *
```

DEFINITIONS --OPCODES

```
189          TTLSS   OPCODES                                                    [05NOV77]
190     *                                                                       [05NOV77]
191     *
192     *    CROSS REFERENCE THESE INSTRUCTIONS WHEN DTSS GMAP          16AUG74 [09DEC79]
193     *                                                                       [09DEC79]
194          OPCREF  DIS,LBAR,LDT,SMIC,CIOC                                    [09DEC79]
195          OPCREF  OPD,OPSYN,DATE                                            [09DEC79]
196          OPCREF  RSW,RMCM,SMCM,LDAC                                        [09DEC79]
197          OPCREF  LDQC,SZNC,LCPR,SCPR                                       [09DEC79]
198          OPCREF  RSCR,SSCR,CIOC,DIS                                        [09DEC79]
199          OPCREF  MLDA,MLDQ,MLDAQ                                           [09DEC79]
200          OPCREF  MSTA,MSTQ,MSTAQ                                           [09DEC79]
201          OPCREF  LMBA,LMBB                                                 [09DEC79]
```

DEFINITIONS --MACHINE CONSTANTS

```
                    202             TTLSS   MACHINE CONSTANTS                                       [05NOV77]
                    203     *
                    204     *       BITS FOR REPEAT INSTRUCTIONS
                    205     *
                    206             HEAD    M               M FOR MACHINE
            002000  207     RTAL    BOOL    2000            TALLY POSITION
            001000  208     ABIT    BOOL    1000            INCREMENT FIRST REGISTER ON RPD
            000400  209     BBIT    BOOL    400             INCREMENT SECOND REGISTER ON RPD
            000200  210     CBIT    BOOL    200             LOAD XO FROM BITS 0-17 OF INSTRUCTION
            000100  211     TZE     BOOL    100             TERMINATE IF ZERO INDICATOR ON
            000040  212     TNZ     BOOL    40              TERMINATE IF ZERO INDICATOR OFF
            000020  213     TMI     BOOL    20              TERMINATE IF NEGATIVE INDICATOR ON
            000010  214     TPL     BOOL    10              TERMINATE IF NEGATIVE INDICATOR OFF
            000004  215     TRC     BOOL    4               TERMINATE IF CARRY INDICATOR ON
            000002  216     TNC     BOOL    2               TERMINATE IF CARRY INDICATOR OFF
            000001  217     TOV     BOOL    1               TERMINATE IF OVERFLOW INDICATOR ON
                    218     *
                    219     *       BITS WE USE IN INDICATOR REGISTER
                    220     *
            004000  221     OVMSK   BOOL    4000            MASK TO STOP FAULT ON OVERFLOW
            001000  222     PERR    BOOL    1000            BIT INDICATING A PARITY ERROR
            000400  223     PMSK    BOOL    400             MASK TO STOP FAULT ON PARITY ERROR
            000200  224     MMODE   BOOL    200             MASTER MODE BIT
            000040  225     MWRD    BOOL    40              MULTI-WORD INTERRUPT INDICATOR
            000001  226     SQUEZ   BOOL    1               NOT A REAL INDICATOR BIT, SAYS JOB IS SQUEEZED
                    227     *
                    228     *       DCW MANIPULATION BITS
                    229     *
            000000  230     IOTD    BOOL    000000          IOTD ACTION CODE FOR DCWS               [08AUG77]
            010000  231     IOTP    BOOL    010000          IOTP ACTION CODE FOR DCWS               [08AUG77]
            020000  232     TDCW    BOOL    020000          TDCW ACTION CODE                        [08AUG77]
            030000  233     IONTP   BOOL    030000          IONTP ACTION CODE                       [08AUG77]
                    234                                                                             [08AUG77]
            020000  235     IONPB   BOOL    020000          BIT TO TELL THE DIFFERENCE BETWEEN IOTP AND IONTP [08AUG77]
                    236                                                                             [08AUG77]
            040000  237     NCB     BOOL    40000           NO CHANGE BIT FOR LPW
            700000  238     IDCW    BOOL    700000          IDCW IDENTIFICATION BITS                [05NOV77]
            040000  239     EC      BOOL    040000          IDCW BIT TO CHANGE THE ADDRESS EXTENSION [05NOV77]
            020000  240     CONT    BOOL    020000          IDCW BIT TO SPECIFY COMMAND CHAINING    [05NOV77]
                    241     *
                    242     *
                    243     *       PUB AND DEVICE DEFINITIONS, MOSTLY FOR IOM USE
                    244     *
            000174  245     PBMAX   EQU     31*4            MAX PUB NUMBER, ADJUSTED FOR I/O
            000040  246     PBPAY   EQU     8*4             FIRST PAYLOAD CHANNEL ON IOM
            000174  247     PBMSK   EQU     31*4            MASK FOR ADJUSTED PUB FIELD
            000077  248     DVMAX   EQU     63              LARGEST PERMISSIBLE DEVICE NUMBER
```

```
M                                DEFINITIONS --MACHINE CONSTANTS

                    249          EJECT                                                                  [09DEC79]
                    250  *
                    251  *       MASK FOR THE OP CODE FIELD
                    252  *
          777000    253  OPCD    BOOL    777000
          777700    254  TLMK    BOOL    777700          TALLY FIELD MASK
          000100    255  TAL1    BOOL    000100          TALLY OF ONE
                    256  *
                    257  *       BITS RETURNED FROM RSW 2 INSTRUCTION
                    258  *
          030000    259  PTYPE   BOOL    030000          (UPPER) PROCESSOR TYPE MASK
          010000    260  DPSE    BOOL    010000          (UPPER) TYPE CODE FOR DPS-E PROCESSORS
          020000    261  ELS     BOOL    020000          (UPPER) TYPE CODE FOR ELS PROCESSORS
          007740    262  FTVMK   BOOL    007740          (UPPER) FAULT VECTOR SWITCHES
                    263
          040000    264  TSOPT   BOOL    040000          (LOWER) TIME-SHARING OPTION (66/X7)                    [30DEC76]
          004000    265  NSA     BOOL    004000          (LOWER) PROCESSOR IS IN VIRTUAL MODE
          002000    266  SER66   BOOL    002000          (LOWER) PROCESSOR IS A SERIES 66
          001000    267          BOOL    001000
          000400    268  CACHE   BOOL    000400          (LOWER) PROCESSOR HAS CACHE MEMORY INSTALLED
          000200    269  EXMEM   BOOL    000200          (LOWER) PROCESSOR HAS EXTENDED MEMORY OPTION
          000100    270          BOOL    000100
          000040    271          BOOL    000040
          000020    272  EIS     BOOL    000020          (LOWER) PROCESSOR HAS EIS INSTALLED
          000010    273  SLMEM   BOOL    000010          (LOWER) PROCESSOR HAS SLOW MEMORY
          000004    274  OVRLP   BOOL    000004          (LOWER) PROCESSOR HAS NO OVERLAP
          000003    275  PROCN   BOOL    000003          (LOWER) PROCESSOR NUMBER
                    276
                    277          UPPER   FTVMK
                    278          LOWER   TSOPT,NSA,SER66,CACHE,EXMEM,EIS,SLMEM,OVRLP,PROCN                [30DEC76]
                    279  *                                                                                [22SEP78]
                    280  *       SPECIAL ADDRESSES USED TO READ SCU REGISTERS (WITH RSCR INSTRUCTION)      [22SEP78]
                    281  *                                                                                [22SEP78]
          000010    282  SCFIG   BOOL    000010          CONFIGURATION SWITCHS                             [22SEP78]
          000040    283  SCCLK   BOOL    000040          REAL-TIME CLOCK                                   [22SEP78]
          000060    284  SCMDR   BOOL    000060          STORE UNIT MODE REGISTER                          [22SEP78]
                    285  *        CSS FIELDS DEFINED IN THE STORE UNIT MODE REGISTER (RETURNED FROM RSCR 60)  * CSS
                    286  *        * CSS
          007760    287  SYNDR   BOOL    7760            SYNDROME -- NON-ZERO IF AN EDAC HAS OCCURRED  ***CSS
          000017    288  STUID   BOOL    17              STORE UNIT ID (0=1K , 17=4K MOS , OTHER=CORE)   ***CSS
                    289  *                                                                                [05NOV77]
```

```
M                       DEFINITIONS --MACHINE CONSTANTS

                  290           EJECT
                  291   *                                                                    [05NOV77]
                  292   *       FAULT NUMBERS                                                [05NOV77]
                  293   *                                                                    [05NOV77]
000000            294   SHD     EQU     0               SHUTDOWN FAULT                       [05NOV77]
000001            295   MEM     EQU     1               MEMORY (STORE) FAULT                 [05NOV77]
000002            296   MME     EQU     2               MME FAULT                            [05NOV77]
000003            297   FAU     EQU     3               FAULT TAG FAULT                      [05NOV77]
000004            298   TIM     EQU     4               TIMER RUNOUT FAULT                   [05NOV77]
000005            299   CMD     EQU     5               COMMAND FAULT                        [05NOV77]
000006            300   DRL     EQU     6               DERAIL FAULT                         [05NOV77]
000007            301   LOK     EQU     7               LOCKUP FAULT                         [05NOV77]
000010            302   CON     EQU     8               CONNECT FAULT                        [05NOV77]
000011            303   PAR     EQU     9               PARITY FAULT                         [05NOV77]
000012            304   IPR     EQU     10              ILLEGAL PROCEDURE (ZOP) FAULT        [05NOV77]
000013            305   ONC     EQU     11              OP-NOT-COMPLETE FAULT                [05NOV77]
000014            306   SUF     EQU     12              STARTUP FAULT                        [05NOV77]
000015            307   OVE     EQU     13              OVERFLOW FAULT                       [05NOV77]
000016            308   DIV     EQU     14              DIVIDE CHECK FAULT                   [05NOV77]
000017            309   EXE     EQU     15              EXECUTE FAULT                        [05NOV77]
                  310   *
                  311   *       DEFINE SIZE OF CPU FAULT VECTOR
                  312   *
000040            313   FVSIZ   BOOL    40              SIZE OF CPU FAULT VECTOR (16 FAULT TYPES)
```

M DEFINITIONS --STATE VECTORS

```
                    314          TTLSS   STATE VECTORS
                    315          HEAD    S                FOR STATE VECTOR
                    316
          000000    317    REG     EQU     0               REGISTERS
          000010    318    AREG    EQU     REG+8           EIS ADDRESS REGISTERS                                  [08AUG77]
          000020    319    PTLEN   EQU     AREG+8          EIS POINTER AND LENGTH REGISTERS                       [08AUG77]
          000030    320    IC      EQU     PTLEN+8         INSTRUCTION COUNTER/INDICATOR REGISTER                 [08AUG77]
          000031    321    BARS    EQU     IC+1            RELATIVE BAR IN SQUEEZE MODE
          000032    322    TACES   EQU     BARS+1          UPPER -- STATUS WORD ONE FOR TRAP OF RUN
          000032    323    LIMIT   EQU     TACES           LOWER -- MAX CORE LIMIT IMPOSED
          000033    324    TIMER   EQU     LIMIT+1         TIMES FOR USER USE
          000034    325    UTYPE   EQU     TIMER+1         UPPER -- USER TYPE                                     [01FEB77]
                    326                                    LOWER -- SPARE                                         [01FEB77]
          000035    327    FTYPE   EQU     UTYPE+1         UPPER -- FAULT TYPE CODE OF LAST FAULT
          000035    328    BUSY    EQU     FTYPE           LOWER -- NUMBER OF TRAPS OUTSTANDING
          000036    329    SWAP    EQU     BUSY+1          UPPER -- NO. TIMES JOB HAS SWAPPED
          000036    330    FRUN    EQU     SWAP            LOWER -- FILE REFERENCE # IN SUPRA JOB
          000037    331    QUANT   EQU     FRUN+1          MAX RUNTIME IMPOSED BY SUPRA JOB
          000040    332    JACES   EQU     QUANT+1         JOB ACCESS IMPOSED BY SUPRA JOB
          000041    333    CATW    EQU     JACES+1         MAX WORDS ALLOWED TO ADD TO CATALOGED FILES
          000042    334    SCRW    EQU     CATW+1          MAX WORDS ALLOWED TO ADD TO SCRATCH FILES
          000043    335    JTIME   EQU     SCRW+1          TIME ACCUMULATED BY TERMINATED INFRA JOBS
          000044    336    TCPU    EQU     JTIME+1         ACTUAL CPU TIME ACCUMULATED (INC. TERM. INFRAS)        [04JUL77]
          000045    337    TIO     EQU     TCPU+1          I/O UNITS ACCUMULATED (INCLUDING TERMINATED INFRAS)    [04JUL77]
          000046    338    TCORE   EQU     TIO+1           CORE UNITS ACCUMULATED (INC. TERM INFRAS)              [04JUL77]
          000047    339    INTP    EQU     TCORE+1         UPPER -- POINTER TO LAST TRAP/FAULT GIVEN              [04JUL77]
          000047    340    BIT     EQU     INTP            LOWER -- BIT CODED INFO (SEE B$ BITS)
          000050    341    JMEM    EQU     BIT+1           REQUESTED LEN/LOC OF JOB
          000051    342    SVMEM   EQU     JMEM+1          REQUESTED LEN/LOC OF STATE VECTOR
          000052    343    STIME   EQU     SVMEM+1         TIMER VALUE LAST PLACED IN S$REG+7
          000053    344    IOCHG   EQU     STIME+1         IO UNITS ON THIS QUANTUM
          000054    345    CORET   EQU     IOCHG+1         CORE QUANTUM TIMER
          000055    346    CLEN    EQU     CORET+1         LENGTH OF JOB FOR CORE SIZE CHARGES
          000056    347    SPEC    EQU     CLEN+1          HEAD/TAIL FCBS OF STACKED SPECIALS                     [17OCT76]
          000057    348    HOLE    EQU     SPEC+1          UPPER -- TOTAL SPACE OCCUPIED BY HOLES                 [01FEB77]
                    349                                    LOWER -- SPARE                                         [01FEB77]
          000060    350    FREE    EQU     HOLE+1          LINK HEADER TO FREE STORAGE LIST                       [17OCT76]
                    351    *       * CSS **
                    352    *        WAS   FCB:EQU FREE+1"LENGTH OF FR LIST / LENGTH OF STATE VECTOR               [04JUL77]
          000061    353    CPFAC   EQU     FREE+1          CP USAGE FACTOR
          000062    354    IOFAC   EQU     CPFAC+1         IO UNITS FACTOR
          000063    355    SFAC    EQU     IOFAC+1         SCHEDULING FACTOR
          000064    356    PID     EQU     SFAC+1          PROJECT ID (3 WORDS)
          000067    357    UMPY    EQU     PID+3           UNITS MULTIPLIER (ROYALTY)
          000070    358    IOUCH   EQU     UMPY+1          IO CHARGES ACCUMULATED
          000071    359    IOTIM   EQU     IOUCH+1         IO CORE LOCKOUT TIME
          000072    360    PTIMR   EQU     IOTIM+1         PREVIOUS TIMER WHEN USER LAST INTERRUPTED
          000073    361    FCB     EQU     PTIMR+1         LENGTH OF FR LIST / LENGTH OF STATE VECTOR             [04JUL77]
                    362    *       *  CSS  **
          000074    363    FR      EQU     FCB+1           FCB / INDEX -> LOC/LEN
                    364                                    LOC IS RELATIVE TO $STATE
                    365    *                                                                                     [09DEC79]
```

S

DEFINITIONS --STATE VECTORS

```
                366             UPPER   TACES,UTYPE,FTYPE,SWAP,INTP,HOLE                [09DEC79]
                367             LOWER   LIMIT,BUSY,FRUN,BIT                             [09DEC79]
```

S DEFINITIONS --FILE CONTROL BLOCKS

```
                    368             TTLSS   FILE CONTROL BLOCKS                                                     [09DEC79]
                    369     *                                                                                       [09DEC79]
                    370             HEAD    F
          000000    371     ACC     EQU     0               UPPER - ACCESS WITH WHICH FILE IS OPEN
          000000    372     TYPE    EQU     0               LOWER - PREFERENCE / TYPE
          000001    373     ABORT   EQU     1               UPPER - WHEN BUSY, PTR TO CONTROLLING LIST EL
                    374                                             IF SIGNED, PTR TO BLOCK FOR RESET STATUS
                    375                                             IF MESSAGE EXISTS, PTR TO SUPRA TRAP
          000001    376     RET     EQU     1               LOWER - FRN IN SUPRA JOB IF A PASSED FILE
                    377                                             IF A BUSY CM, FRN AT CURRENT SLAVE END
          000002    378     CAT     EQU     2               IF REGULAR CATALOGED FILE --
          000000    379                                     UPPER - POINTER TO C$USE LIST ENTRY
                    380                                     LOWER - MAXIMUM ACCESSES ALLOWED ON FILE
          000000    381                                     IF SHARED FILE --
          000000    382                                     UPPER - FRN OF DATA FCB + SIGN BIT
          000000    383                                     LOWER - MAXIMUM ACCESSES ALLOWED ON FILE
                    384                                     IF COMUNICATIONS FILE --
                    385                                     UPPER -- POINTER TO THE TABLE ENTRY FOR THE MASTE
                    386                                     LOWER -- POINTER TO THE TABLE ENTRY FOR THIS END
          000003    387     M       EQU     3               UPPER -- IF A BUSY CS, TRAP LOCATION
                    388                                             IF A BUSY JOB FILE, TRAP LOCATION
                    389                                             IF IN CS COPY, FR OF COMM FILE
                    390                                             IF MESSAGE EXISTS, LOC IN SUPRA
                    391                                     LOWER - IF A BUSY CM, LENGTH LEFT TO COPY
                    392                                             IF A BUSY CS, FRN OF OTHER END
                    393                                             IN INITIAL SWAP FCB, FRN OF SUPRA JOB
                    394                                             IF MESSAGE EXISTS, MESSAGE LENGTH
          000004    395     POINT   EQU     4               FILE POINTER
                    396                                     WHEN DEST. FILE, DEBIT TO S$SCRW OR S$CATW
          000005    397     N       EQU     5               FILE LENGTH OR CATALOG MAX
                    398                                     ZERO FOR JOB FILES MEANS JOB TERM'D
          000006    399     DA      EQU     6               FIRST DEVICE ADDRESS
                    400     *
                    401     *       FORMAT OF DATA FCB FOR SHARED FILES
                    402     *
                    403     ACC             0               ACCESSES = 0
                    404     TYPE            0               TYPE OF FILE BEING SHARED IN PREF FIELD
          000000    405                                     TYPE FIELD = F$DATA
                    406     ABORT           1               ABORT = B$SIGN (NOT ABORTABLE)
                    407     RET             1               RET = TRAP LOCATION FOR MME
                    408     CAT             2               UPPER - POINTER TO MASTER C$USE
          000000    409                                     LOWER - POINTER TO OUR C$USE
          000003    410     PT1     EQU     3               SOURCE FILE POINTER SAVED HERE
          000004    411     PT2     EQU     4               DEST FILE POINTER SAVED HERE
          000005    412     CPL     EQU     5               COPY LENGTH
          000006    413     SFR     EQU     6               UPPER - FRN OF SOURCE FILE OF COPY
          000006    414     DFR     EQU     SFR             LOWER - FRN OF DESTINATION FILE
          000007    415     WDC     EQU     7               MAX DEBIT TO CATW
          000010    416     DATL    EQU     8               LENGTH OF DATA FCB
                    417     *                                                                                       [09DEC79]
                    418             UPPER   ACC,ABORT,SFR                                                           [09DEC79]
                    419             LOWER   TYPE,RET,DFR                                                            [09DEC79]
```

F DEFINITIONS --FILE CONTROL BLOCKS

```
                    420             EJECT
                    421     *
                    422     *           F$TYPE CONTAINS THE FILE PREFERENCE IN BITS 26-31 IN THE
                    423     *       FOLLOWING FORMAT.   BITS 26-28 UNUSED.
                    424     *                           BITS 29-31  PREFERENCE SPECIFIED BY CREATOR
                    425     *       THE FILE TYPE IS SPECIFIED IN BITS 32-35 BY THE FOLLOWING CODES.
                    426     *
000017              427     TYP     BOOL    17              TYPE FIELD MASK
000001              428     CTF     BOOL    1               A CATALOG AS A FILE                     16AUG74
000002              429     CTL     BOOL    2               A CATALOGED FILE OR CATALOG             16AUG74
000000              430     RSF     BOOL    0               REGULAR SCRATCH FILE
000001              431     RSC     BOOL    1               REGULAR SCRATCH CATALOG
000002              432     RCF     BOOL    2               REGULAR CAT FILE
000003              433     RCC     BOOL    3               REGULAR CAT CAT
000004              434     SPE     BOOL    4               SPECIAL FILES                           16AUG74
000004              435     SSF     BOOL    4               SPECIAL SCRATCH FILE
000005              436     SSC     BOOL    5               SPECIAL SCRATCH CAT
000006              437     SCF     BOOL    6               SPECIAL CAT FILE (DEVICE FILE)
000007              438     SCC     BOOL    7               SPECIAL CAT CAT
000010              439     SPL     BOOL    10              OTHER SPECIAL TYPES                     16AUG74
000010              440     CS      BOOL    10              COMUNICATIONS SLAVE
000011              441     CM      BOOL    11              COMMUNICATIONS MASTER
000012              442     RJ      BOOL    12              RUNNING JOB
000013              443     NF      BOOL    13              NONEXISTANT FILE
000014              444     OFL     BOOL    14              OFF LINE FILE
000015              445     DATA    BOOL    15              SHARED MODE OR STACKED SPECIAL DATA FCB                 [17OCT76]
000016              446     SHR     BOOL    16              SHARED MODE FILE (NOTE: CTL ON, CTF OFF)
000014              447     SPT     BOOL    14              ALL SPECIAL FILE TYPES                  16AUG74
                    448     *
000020              449     PREF    BOOL    20              FILE PREFERENCE FIELD MULTIPLIER                        [09DEC79]
000160              450     PREFM   BOOL    160             PREFERENCE FIELD MASK
200000              451     PASSD   BOOL    200000          FLAG FOR PASS-IN-PROGRESS
400000              452     ALT     BOOL    400000          FLAG BIT TO INDICATE FILE ALTERED
100000              453     LALT    BOOL    100000          LENGTH OF THE FILE HAS BEEN ALTERED
```

F DEFINITIONS --CATALOG SYMBOLS

```
                    454             TTLSS   CATALOG SYMBOLS
                    455     *                                                                       [09DEC79]
                    456             HEAD    C                                                       [09DEC79]
                    457     *                                                                       [09DEC79]
                    458     *       CATALOG HEADER                                                  [09DEC79]
                    459     *                                                                       [09DEC79]
000000              460     MAX     EQU     0               MAXIMUM TOTAL FILE LENGTH
000001              461     ALOC    EQU     1               ACTUAL TOTAL FILE LENGTH
000002              462     TRAN    EQU     2               *-* NON-ZERO IF CAT HAS TRANSPARENT SUBCATS
                    463     *       *-*  NOTE: WORD TWO HAS SPECIAL MEANING FOR CORE CATS.
000003              464     CKSUM   EQU     3               HEADER CHECKSUM FOR A NON-CORE CATALOG
                    465                                     FOR THE MFD, A CHECKSUM OF THE ENTIRE
                    466                                     CATALOG IS STORED HERE ON A SOFT CRASH
000004              467     CLEN    EQU     4               UPPER - LEN OF DA LIST
000005              468     INDEX   EQU     5               UPPER - POINTER TO FIRST ENTRY
000005              469     ENTRY   EQU     5               LOWER - NUMBER OF ENTRIES
000006              470     CN      EQU     6               LENGTH OF CAT
000007              471     CDA     EQU     7               DA LIST OF CAT
                    472     *                                                                       [09DEC79]
                    473     *       CATALOG ENTRY                                                   [09DEC79]
                    474     *                                                                       [09DEC79]
000000              475     LEN     EQU     0               UPPER - LENGTH OF ENTIRE FCB
000001              476     NAME    EQU     1
000003              477     PASS    EQU     3
000005              478     DATE    EQU     5               UPPER - DATE LAST USED
000006              479     DALT    EQU     6               UPPER - DATE LAST ALTERED
000006              480     TYPE    EQU     6               LOWER- MISC INF
000007              481     ACC     EQU     7               ACCESS INFORMATION
000010              482     N       EQU     8               LENGTH OF FILE OR MAX OF A CAT
000011              483     DA      EQU     9               FIRST DA OF THE FILE
                    484     *
                    485             UPPER   CLEN,INDEX,DALT                                         [09DEC79]
                    486             LOWER   ENTRY,TYPE                                              [09DEC79]
                    487     *
                    488     *       * CSS **
                    489     *       CSS MULTI BUFFERED CATALOG MODULE
                    490     *       POSITIONAL POINTER TABLE
000004              491     PPLEN   EQU     4               NUMBER OF WORDS PER ENTRY IN TABLE
000000              492     PPPDA   EQU     0               PDA
000001              493     PPSEG   EQU     1               SEGMENT NUMBER
000002              494     PPPTR   EQU     2               HISTORY POINTER
000003              495     BFPTR   EQU     3               BUFFER POINTER
                    496     *       HISTORY TABLE
000004              497     NENT    EQU     4               NUMBER OF WORDS PER ENTRY IN TABLE
000000              498     HCNT    EQU     0               HISTORY COUNTER
000001              499     HPDA    EQU     1               PDA
000002              500     HSEG    EQU     2               SEGMENT NUMBER
000003              501     HUSE    EQU     3               INDICATES HISTORY IN USE BY PPTAB
                    502     *       HISTORY AND BUFFER SIZES
000014              503     NBUF    EQU     12              *-* NUMBER OF CATALOG BUFFERS
000050              504     NHIS    EQU     40              NUMBER OF CATALOG HISTORIES
                    505     *
```

```
C                       DEFINITIONS --CATALOG SYMBOLS
                506     *       * CSS **
```

C DEFINITIONS --CATALOG SYMBOLS

```
                    507             EJECT                                                                   [09DEC79]
                    508     *                                                                               [09DEC79]
                    509     *       FORMAT OF C$USE LIST ENTRIES WHEN USED FOR CATALOGED FILES
                    510     *
000000              511     USEDA   EQU     0               DA OF SUPRA CATALOG PLUS ENTRY NUMBER
000001              512     USEFL   EQU     1               UPPER - USE FLAGS (FOR SHARED FILES, 1ST LINK)
                    513                                     LOWER -- ATTACHMENT COUNT
000002              514     USEI    EQU     2               LENGTH OF A C$USE LIST ENTRY
                    515     *
                    516     *       FORMAT OF C$USE ENTRIES WHEN USED AS COMFILE TABLE ENTRIES
                    517     *       OR SHARED FILE ENTRIES
                    518     *
                    519             HEAD    F
                    520     *
000000              521     BIT     EQU     0               UPPER -- BITS FOR THE COMFILE (BITS 1-11)
                    522                                     BIT 0 IS ALWAYS ON AND BITS 12-17 ARE ALWAYS OFF.
000000              523     LINK    EQU     0               LOWER -- LINK TO NEXT END OF THE FILE
000000              524                                     FOR COMFILES -
                    525                                     THE MASTER END IS LINKED TO THE LOWEST SLAVE END,
                    526                                     SLAVE ENDS ARE LINKED IN INCREASING ORDER,
                    527                                     THE HIGHEST SLAVE IS LINKED TO THE MASTER.
000001              528     J       EQU     1               UPPER -- JOB NUMBER AT THIS END
000001              529     FR      EQU     1               LOWER -- FRN OF THE COMFILE AT THIS END
                    530     *                                                                               [09DEC79]
                    531             UPPER   BIT,J                                                           [09DEC79]
                    532             LOWER   LINK,FR                                                         [09DEC79]
```

F

DEFINITIONS --B$ BITS

```
                    533             TTLSS   B$ BITS                                                                 [09DEC79]
                    534                                     GENERAL BIT DEFINITIONS                                 [09DEC79]
                    535             HEAD    B                                                                       [09DEC79]
          400000    536   SIGN      BOOL    400000          SIGN BIT                                                [09DEC79]
          040000    537   TRAPE     BOOL    040000          E REGISTER FLAG FOR SLAVE TRAP PROGRAMS                 [09DEC79]
                    538   *                                                                                       [09DEC79]
          002000    539   MKS       BOOL    2000            MULTIPLE OF K CURRENTLY USED IN SLAVE MEM
          001000    540   SSV       BOOL    1000            STANDARD SV SIZE
          001000    541   LSV       BOOL    1000            LARGE SV SIZE
          000040    542   MXFCB     EQU     32              MAX LENGTH OF AN FCB *****MAY BE INCOMPLETELY PARAMATERIZED
          000100    543   UTYPS     EQU     64              NUMBER OF VALID USER TYPES                              [01FEB77]
          000100    544   MXSV      EQU     64              MAX NUMBER OF FRNS FOR SMALL STATE VECTOR               [17OCT76]
          000036    545   MXLG      EQU     30              MAX LENGTH TO LOG ON LOG MME                            [30DEC76]
                    546   *
                    547   *       BIT DEFINITIONS FOR J$BIT
                    548   *
          400000    549   SWAP      BOOL    400000          SWAP IN PROGRESS
          200000    550   CORE      BOOL    200000          1 IF IN CORE, 0 IF NOT
          100000    551   TERM      BOOL    100000          1 - IN PROCESS OF TERMINATE / NOT
          040000    552   CONA      BOOL    40000           IF 1 MUST BE RESET BY IMMEDIATE SUPRA
          020000    553   CONB      BOOL    20000           IF 1 MAY BE RESET BY ANY JOB ABOVE
          010000    554   MDA       BOOL    10000           INITIAL SWAPIN IN PROGRESS
          004000    555   SWREQ     BOOL    4000            SWAP-REQUESTED BIT
          002000    556             BOOL    2000                                                                    [09DEC79]
          001000    557   PBLK      BOOL    1000            SWAP IN OF PURE PROCEDURE IN PROGRESS                   [09DEC79]
          000400    558   UPS       BOOL    400             UNSET PAUSE - SWAP BIT                                  [09DEC79]
          000200    559   RUN       BOOL    200             JOB IS RUNNING OR QUEUED TO RUN                         [09DEC79]
          000100    560             BOOL    100                                                                     [09DEC79]
                    561   *         WAS     SPARE:REM 40"SPARE BIT                                                  [08AUG77]
                    562   *       * CSS   **
          000040    563   LIMIT     BOOL    40              JOB IS LIMITED IN CRU USAGE
                    564   *       * CSS   **
          000020    565   TERMB     BOOL    20              SET IF AT BOTTOM OF TERMINATING TREE
          000010    566   TERMC     BOOL    10              TEMPORARY FLAG TO INDICATE FRO OP UNTRAPPED
          000004    567   SWER      BOOL    4               ERROR OCCURRED ON SWAP-IN
          000002    568   COURT     BOOL    2               COURTESY SHOT IN PROGRESS
          000001    569   CLOCK     BOOL    1               CLOCK RUNNING
          414000    570   CPBLK     EQU     SWAP+MDA+SWREQ  BITS WHICH BLOCK COPY SUBROUTINE                         [09DEC79]
```

B DEFINITIONS --B$ BITS

```
                    571             EJECT                                                                   [01SEP79]
                    572     *
                    573     *
                    574     *       BIT DEFINITIONS FOR S$BIT
                    575     *
400000              576     SPINP   BOOL    400000          SPECIAL INTERRUPT IS PENDING
200000              577     ENTRY   BOOL    200000          INITIAL EXEC ENTRY FLAG
100000              578     NRUN    BOOL    100000          JOB RUN WITH "NEW" RUN MME
040000              579     SPOVF   BOOL    040000          SPECIAL INTERRUPT STACK HAS OVERFLOWED
020000              580     TRAPP   BOOL    020000          JOB RUN AS SLAVE TRAP PROGRAM
                    581     *       * WAS   BOOL 010000"SPARE                                               [01SEP79]
                    582     *       * WAS   BOOL 004000"SPARE                                               [01SEP79]
                    583     *       * CSS   **
010000              584     TRAPR   BOOL    10000           SLAVE TRAP PROGENY BIT
004000              585     NPRN    BOOL    4000            NON-PRIVILEGED RUN IN PROGRESS
                    586     *       * CSS   **
002000              587     LPAUS   BOOL    002000          JOB HAS ISSUED A 'LONG PAUSE'                   [09DEC79]
001000              588     FLOG    BOOL    001000          JOB HAS LOG MME OUTSTANDING                     [22JUN76]
                    589     *
                    590     *       BIT DEFINITIONS FOR J$TYPE
                    591     *
400000              592     MON     BOOL    400000          MONITOR
200000              593     CRRES   BOOL    200000          CORE RESIDENT
100000              594     SV      BOOL    100000          LARGE STATE VECTOR
040000              595     CATPR   BOOL    040000          SPECIAL CATALOG PRIVILEGE;ALLOWS:               [09DEC79]
                    596                                     1) CREATE INFINITE MAX CATALOGS
                    597                                     2) PREALLOCATE LARGE SCRATCH CATALOGS
                    598                                     3) CMAX BELOW ALLOC                             [17OCT76]
020000              599     DUMP    BOOL    20000           LOAD-DUMP PRIVILIGE;ALLOWS:
                    600                                     1) NON-SHRINKING STATE VECTOR
                    601                                     2) PROVIDE DEVICE ADDRESS MME
                    602                                     3) PREALLOCATION OF SCRATCH FILES
                    603                                     4) CATALOG WITH CODED DATES, DAYS USED COUNTS
                    604                                     5) CATALOG, OPEN WITHOUT QUOTA CHECKS
010000              605     PDQ     BOOL    10000           EXPRESS SERVICE BIT
                    606     *       * WAS     CRASH:BOOL 4000 "JOB MAY CRASH SYSTEM                         [17OCT76]
                    607     *       * CSS   **
004000              608     PID     BOOL    4000            JOB MAY SET PROJECT ID'S IN RUN LIST
                    609     *       * CSS   **
002000              610     LOG     BOOL    2000            JOB MAY LOG TO CONSOLE                          [17OCT76]
                    611     *                                                                               [17OCT76]
                    612     *       DEFINITIONS FOR S$FTYPE                                                 [17OCT76]
                    613     *                                                                               [17OCT76]
000001              614     MME     EQU     1               JOB ISSUED MME                                  [17OCT76]
000002              615     DIR     EQU     2               JOB GOT FAULT AND HAD A DIRTY FAULT VECTOR      [17OCT76]
000003              616     PAR     EQU     3               A PARITY FAULT OCCURRED WHILE THE JOB WAS EXECUTING [17OCT76]
000004              617     TROF    EQU     4               JOB SHOULD BE RESCHEDULED AND SENT BACK OUT     [17OCT76]
```

B DEFINITIONS --B$ BITS

```
                    618             EJECT
                    619     *
                    620     *       BIT DEFINITIONS FOR F$ACCESS
                    621     *
400000              622     CAT     BOOL    400000          CATALOG BIT
200000              623     BSY     BOOL    200000          BUSY BIT
200000              624     PASS    BOOL    200000          PASSWORD TO BE CHECKED FLAG
200000              625     SHR     BOOL    200000          SHARED BIT FOR ACCESS WORD
100000              626     RET     BOOL    100000          RETURN BIT
040000              627     OWN     BOOL    40000           OWNER
020000              628     PASSD   BOOL    20000           FILE HAS BEEN PASSED
010000              629     EX      BOOL    10000           EXECUTE / SEARCH BIT
004000              630     AP      BOOL    4000            APPEND
002000              631     WT      BOOL    2000            WRITE
001000              632     RD      BOOL    1000            READ
057000              633     ALL     EQU     OWN+EX+AP+WT+RD ALL ACCESSES
000400              634     LS      BOOL    400             USER LIST PERMISSION                    16AUG74
000777              635     UBITS   BOOL    777             USER ACCESS BITS
000400              636     CW      BOOL    400             COMM FILE BIT SAYING SLAVE END DID WRITE
000200              637     CR      BOOL    200             COMM FILE BIT SAYING SLAVE END ISSUED REA
000040              638     TNT     BOOL    40              0 - TRAP -- 1 - NO TRAP
                    639     *
                    640     *       BIT DEFINITIONS FOR F$BIT
                    641     *
                    642                                     BIT ZERO IS ALWAYS ON TO DISTINGUISH THE ENTRY
                    643                                     FROM A CATALOG DA
                    644     *       BOOL    400000          MUST BE ONE
200000              645     CFC     BOOL    200000          COPY IN PROGRESS
100000              646     CFD     BOOL    100000          DRIVE IN PROGRESS
040000              647     CFR     BOOL    40000           SLAVE HAS ISSUED RESET STATUS AND MASTER HAS
                    648                                     NOT ECHOED THE OPERATION
020000              649     CFGA    BOOL    20000           AN OPERATION HAS BEEN ABORTED BY RESET STATUS
010000              650     CFCL    BOOL    10000           THE FILE IS BEING CLOSED
004000              651     CFSCL   BOOL    4000            THIS END REQUESTED A CLOSE
002000              652     CBRKP   BOOL    2000            THIS END HAS BREAK PERMISSION
001000              653     CFRSV   BOOL    1000            FILE IS RESERVED FOR THE SOME SLAVE END
000400              654     CFRVM   BOOL    400             FILE IS RESERVED FOR THIS END
000200              655     CWTNG   BOOL    200             THIS END WAITING FOR SHARED MODE ACCESS
000100              656     CSHRR   BOOL    100             LAST SHARED COPY INITIATED WAS A READ   [14JUL76]
                    657     *       BOOL    40              MUST BE ZERO
                    658     *       BOOL    20              MUST BE ZERO
                    659     *       BOOL    10              MUST BE ZERO
                    660     *       BOOL    4               MUST BE ZERO
                    661     *       BOOL    2               MUST BE ZERO
                    662     *       BOOL    1               MUST BE ZERO
370400              663     CFBSY   EQU     CFCL+CFGA+CFC+CFD+CFR+CFRVM BITS WHICH MAKE THE FILE BUSY
                    664                                     BITS 12-17 ARE ALWAYS ZERO TO DISTINQUISH
                    665                                     THE ENTRY FORM A CATALOG DA
```

B DEFINITIONS --B$ BITS

```
                    666             EJECT                                                                   [09DEC79]
                    667     *
                    668     *       BIT DEFINITIONS FOR X4 FLAGS ON COMFILE MME'S
                    669     *       AND SHARED FILE ACCESSES.
                    670     *
          400000    671     NTPS    BOOL    400000          DON'T TRAP SLAVE SOURCE
          200000    672     RSVS    BOOL    200000          RESERVE SLAVE SOURCE COMFILE
          000400    673     NTPD    BOOL    400             DON'T TRAP DESTINATION                  16AUG74
          000200    674     RSVD    BOOL    200             RESERVE DESTINATION COMFILE             16AUG74
          177177    675     X4ILL   EQU     -1-NTPS-RSVS-NTPD-RSVD ILLEGAL BITS
                    676     *                                                                               [22JUN76]
                    677     *       BIT FLAGS IN X4 FOR LOG MME                                             [22JUN76]
                    678     *                                                                               [22JUN76]
          400000    679     DING    BOOL    400000          RING CONSOLE ALARM                              [22JUN76]
          200000    680     FILE    BOOL    200000          LOG TO FILE ONLY                                [22JUN76]
                    681     *
                    682     *       BIT DEFINITIONS FOR C$USE
                    683     *
          400000    684     CAP     BOOL    400000          APPEND FLAG FOR C$USE
          200000    685     CWT     BOOL    200000          WRITE FLAG FOR C$USE
          100000    686     HLD     BOOL    100000          FILE IS HELD OPEN BY EXEC
          040000    687     CSHR    BOOL    040000          SHARED MODE FLAG FOR C$USE
          600002    688     CALL    EQU     CWT+CAP+2       ALL ATTCHMENTS ON FILE
          200003    689     CCAL    EQU     CWT+3           ALL ATTCHMENTS ON CATALOG
                    690     *
                    691     *       BIT DEFINITIONS FOR C$USE FLAGS
                    692     *
          400000    693     DLU     BOOL    400000          DLU IS NOT CURRENT
          200000    694     DLA     BOOL    200000          DLA IS NOT CURRENT
          100000    695     ALT     BOOL    100000          FILE HAS BEEN ALTERED
          040000    696     APREL   BOOL    40000           APPEND HAS BEEN RELEASED
          020000    697     LALT    BOOL    20000           LENGTH HAS BEEN ALTERED
          007777    698     CNT     BOOL    7777            MASK FOR HOLD COUNT FIELD
```

B DEFINITIONS --B$ BITS

699 EJECT [09DEC79]

```
                700     *
                701     *       BIT DEFINITIONS FOR C$ACCESS
                702     *
400000          703     TRAP    BOOL    400000          FILE IS PROTECTED BY A TRAP PROGRAM
200000          704     STRAP   BOOL    200000          FILE IS PROTECTED BY A SLAVE TRAP PROGRAM
020000          705     CLI     BOOL    20000           FILE IS ACCESSABLE THROUGH CLIMBING
020000          706     TLESC   BOOL    20000           SPECIAL SCAN CONVENTIONS ARE ILLEGAL
000400          707     TVAL    BOOL    400             EXEC TRAP BIT FOR VALIDATION FILES
000200          708     TMUD    BOOL    200             EXEC TRAP BIT FOR MASTER USER DIRECTORY
000100          709     TBIL    BOOL    100             EXEC TRAP BIT FOR BILLING CATALOG
000040          710     TPER    BOOL    40              EXEC TRAP BIT FOR PERIPHERALS AND SUCH
000020          711     TBAK    BOOL    20              EXEC TRAP BIT FOR BACKGROUND JOB CATS
000010          712     TLIB    BOOL    10              EXEC TRAP BIT FOR LIBRARIES
000004          713     TMOS    BOOL    4               EXEC TRAP BIT FOR MONITOR/SYSTEMS CATS
000002          714     TMFD    BOOL    2               EXEC TRAP BIT FOR MASTER FILE DIRECTORY
000001          715     TSYS    BOOL    1               EXEC TRAP BIT FOR SYSTEMS PROGRAMMERS
000777          716     TALL    BOOL    777             ALL EXEC TRAP BITS
                717     *
                718     *       BIT DEFINITIONS FOR SPECIAL INTERRUPTS
                719     *
000000          720     STM     EQU     0               SET MODE
000001          721     RCF     EQU     1               READ COMMUNICATION FIRL
000002          722     SRS     EQU     2               SLAVE RESET STATUS ON COMM. FILE
000003          723     BRK     EQU     3               BREAK
000004          724     PF      EQU     4               PASS FILE
000005          725     RF      EQU     5               RETURNED FILE
000006          726     FCLO    EQU     6               FILE CLOSED
000010          727     ITA     EQU     8               ILLEGAL TRAP ADDRESS
000012          728     D       EQU     10              DRIVE
000013          729     R       EQU     11              READ
000014          730     W       EQU     12              WRITE
000015          731     RS      EQU     13              REQUEST STATUS
000016          732     T       EQU     14              TRUNCATE
000017          733     SP      EQU     15              SET POINTER
000024          734     MRD     EQU     20              MULTI-RECORD DRIVE (DCWS)       [15DEC76]
```

B DEFINITIONS --B$ BITS

```
                    735             EJECT                                                                   [09DEC79]
                    736     *
                    737     *       GENERAL BIT DEFINITIONS FOR STATUS RETURNS                      16AUG74
                    738     *
000000              739     OK      BOOL    0               OK
000020              740     STAT    BOOL    20              STATUS WAS RESET
000040              741     QUEX    BOOL    40              QUOTAS EXCEEDED
000060              742     NSTOR   BOOL    60              SYSTEM OUT OF STORAGE
000100              743     ACER    BOOL    100             ACCESS ERROR
000120              744     BUSY    BOOL    120             FILE BUSY
000140              745     ERRA    BOOL    140             A REGISTER PARAMETER ERROR
000160              746     ERRQ    BOOL    160             Q REGISTER PARAMETER ERROR
000200              747     ERR0    BOOL    200             X0 PARAMETER ERROR
000220              748     ERR1    BOOL    220             X1 PARAMETER ERROR
000240              749     ERR2    BOOL    240             X2 PARAMETER ERROR
000260              750     ERR3    BOOL    260             X3 PARAMETER ERROR
000300              751     ERR4    BOOL    300             X4 PARAMETER ERROR
000320              752     ERR5    BOOL    320             X5 PARAMETER ERROR
000360              753     ERR7    BOOL    360             X7 PARAMETER ERROR
000400              754     RERR    BOOL    400             RECOVERABLE ERROR
000420              755     UERR    BOOL    420             UNRECOVERABLE ERROR
000440              756     UNABL   BOOL    440             NOT SUFFICIENTLY ENABLED                        [01MAY79]
```

B DEFINITIONS --B$ BITS

```
                    757             EJECT                                                                   16AUG74
                    758     *                                                                               16AUG74
                    759     *       SPECIFIC BIT DEFINITIONS FOR STATUS RETURNS                             16AUG74
                    760     *                                                                               16AUG74
                    761     *       COPY,WRITE,READ,DRIVE,READ CAT,READ CAT AND OPEN                        16AUG74
                    762     *                                                                               16AUG74
000001              763     SFE     BOOL    1               SOURCE FILE EXHAUSTED                           16AUG74
000002              764     DFE     BOOL    2               DESTINATION FILE EXHAUSTED                      16AUG74
000003              765     COMP    BOOL    SFE-DFE         (CONSTANT FOR SWITCHING STATUS)                         [15DEC76]
000003              766     INA     BOOL    3               OPERATION INAPPROPRIATE                         16AUG74
000004              767     SPT     BOOL    4               SOURCE POINTER OUT OF BOUNDS                    16AUG74
000005              768     DPT     BOOL    5               DESTINATION POINTER OUT OF BOUNDS               16AUG74
000006              769     CFB     BOOL    6               COMFILE BUSY                                    16AUG74
000007              770     NAS     BOOL    7               OTHER END NOT ACCEPTING SPECIALS                16AUG74
000010              771     BDW     BOOL    10              BAD DCW                                         16AUG74
000014              772     SVF     BOOL    14              STATE VECTOR FULL                               16AUG74
000015              773     JSW     BOOL    15              JOB SWAPPED OUT OF CORE                         16AUG74
                    774     *                                                                               16AUG74
                    775     *       CATALOG                                                                 16AUG74
                    776     *                                                                               16AUG74
000001              777     ITP     BOOL    1               ILLEGAL TRAP PROTECTION                         16AUG74
000014              778     TPL     BOOL    14              PREFERENCE OF SCRATCH FILE TOOO LOW                     [01MAY79]
                    779     *                                                                               16AUG74
                    780     *       OPEN,ERASE,REPLACE
                    781     *                                                                               16AUG74
000001              782     PRL     BOOL    1               PARTIAL SUCCESS                                 16AUG74
000001              783     CTF     BOOL    1               X2 POINTS TO CATALOGED FILE (FROM CATL)
000002              784     PRE     BOOL    2               PREFERENCE TOO LOW  (FROM CATL)
000002              785     LOK     BOOL    2               LOCKOUT
000003              786     NOF     BOOL    3               FILE NOT FOUND
000004              787     PRV     BOOL    4               PROTECTION VIOLATION
000005              788     FAL     BOOL    5               FAIL ON CLIMB
000006              789     BTN     BOOL    6               BAD TREE NAME
000007              790     CLE     BOOL    7               CLIMB ERROR
000010              791     OFL     BOOL    10              OFF LINE FILE
000011              792     SPR     BOOL    11              SPECIAL FILE CANNOT BE REMOVED
000012              793     FER     BOOL    12              FORMAT ERROR
                    794     *
                    795     *       FORMAT ERROR SUB-STATUSES
                    796     *
000001              797     LNG     EQU     1               NAME TOO LONG
000002              798     AST     EQU     2               TOO MANY ASTERISKS IN *** SCAN
000003              799     MUD     EQU     3               ILLEGAL USER NUMBER FORMAT
000004              800     DPS     EQU     4               DUAL PASSWORD
000005              801     CHR     EQU     5               ILLEGAL CHARACTER
000006              802     ILF     EQU     6               ILLEGAL FORMAT
000007              803     ILC     EQU     7               ILLEGAL CONVENTION (CONVENTIONS ARE DISALLOWED)
000010              804     ILS     EQU     8               ILLEGAL USE OF "/" CONVENTION (CATALOG,REPLACE)         [09DEC79]
                    805     *                                                                               [09DEC79]
                    806     *       CCE,UNCAT                                                               16AUG74
                    807     *                                                                               16AUG74
000001              808     FNC     BOOL    1               FILE NOT CATALOGED                              16AUG74
```

B DEFINITIONS --B$ BITS

000002	809	DUP	BOOL	2	DUPLICATE FILENAME	16AUG74	
000003	810	ITB	BOOL	3	ILLEGAL TRAP BITS	16AUG74	
000013	811	IUD	BOOL	13	ILLEGAL USAGE, DATES, PREF, OR TYPE		
	812	*				16AUG74	
	813	*	OTHERS			16AUG74	
	814	*				16AUG74	
000001	815	MXL	BOOL	1	SPECIFIC MAX LESS THAN CURRENT	16AUG74	
000001	816	OVJ	BOOL	1	OVERLAY -- OVERLAY REJECTED	16AUG74	
000002	817	OVM	BOOL	2	OVERLAY -- INSUFFICIENT MEMORY		
000001	818	SLL	BOOL	1	TRUNCATE -- SPEC. LEN LONGER THAN CURRENT	16AUG74	
000001	819	RIP	BOOL	1	RESET STATUS -- ONE ALREADY IN PROGRESS	16AUG74	
000001	820	TRO	BOOL	1	TIMER RUNOUT ON A RUN		
000002	821	ABO	BOOL	2	JOB ABORTED (STATUS RETURN ON A RUN)		
000002	822	MNR	BOOL	2	MESSAGE NOT READ		
000002	823	FCL	BOOL	2	RELINQUISH -- FILE CLOSED	16AUG74	
000005	824	RLE	BOOL	5	RUN LIST ERROR	16AUG74	
000006	825	SWP	BOOL	6	SWAP-OUT ERROR	16AUG74	
000001	826	LG2	BOOL	1	LOG - LOG ALREADY OUTSTANDING		[22JUN76]
000002	827	LGNBF	BOOL	2	LOG - NO BUFFER AVAILABLE		[22JUN76]
000003	828	LGESC	BOOL	3	LOG - ESCAPE SEQUENCE OVERFLOW		[22JUN76]

B DEFINITIONS --LIST ELEMENT SYMBOLS

```
                    829             TTLSS   LIST ELEMENT SYMBOLS
                    830     *
                    831     *               QUEUE MANAGEMENT SYMBOL DEFINITIONS
                    832     *
                    833             HEAD    Q
           000000   834     LINK    EQU     0               QUEUE LINK (UPPER HALF)
           000000   835     RUN     EQU     0               TRANSFER ADDRESS FOR MASTER TASK QUEUE
           000001   836     QLEN    EQU     RUN+1           LENGTH OF ELEMENT HEAD                  [09DEC79]
                    837     *                                                                       [09DEC79]
                    838             UPPER   LINK                                                    [09DEC79]
                    839             LOWER   RUN                                                     [09DEC79]
                    840     *
                    841     *               LIST ELEMENT CONTROL SYMBOL DEFINITIONS
                    842     *
                    843             HEAD    T
           777777   844     LINK    EQU     -1              LINK TO PREVIOUS STORAGE BLOCK
           777777   845     LEN     EQU     -1              LENGTH OF AVAILABLE STORAGE IN BLOCK
                    846     *
                    847             UPPER   LINK                                                    [09DEC79]
                    848             LOWER   LEN                                                     [09DEC79]
                    849     *                                                                       [09DEC79]
                    850     *               INTERRUPT-GENERATED TASK STRUCTURE LIST ELEMENT
                    851     *
                    852             HEAD    X
           000001   853     IENT    EQU     1               IOC QUEUE ENTRY
           000002   854     ILEN    EQU     2               LENGTH OF STRUCTURE
                    855     *
                    856     *
                    857             HEAD    E
           000001   858     XJ      EQU     1               LOC OF X - J IN LIST ELEMENT
```

E DEFINITIONS --PHYSICAL I/O DEFINITIONS

```
                    859             TTLSS   PHYSICAL I/O DEFINITIONS                                                [09DEC79]
                    860     *                                                                                       [09DEC79]
                    861     *                                                                                       [09DEC79]
                    862     *       PHYSICAL IO CALL AND STORAGE BLOCKS                             16AUG74
                    863     *                                                                       16AUG74
                    864             HEAD    I                                                       16AUG74
000001              865     DEV     EQU     1               (UPPER) PHYSICAL UNIT NUMBER
000001              866     TYPE    EQU     1               (LOWER) DEVICE TYPE
000002              867     CMD     EQU     TYPE+1          (UPPER) CURRENT COMMAND TABLE POINTER
                    868                                     (LOWER) SAVED COMMAND TABLE POINTER
000003              869     PUB     EQU     CMD+1           (UPPER) PUB NUMBER *4
000003              870     PUBL    EQU     PUB             (LOWER)                                                 [09DEC79]
000003              871     SPRET   EQU     PUB             (LOWER) SPECIAL TIMEOUT RETURN
000004              872     SEKAD   EQU     PUB+1           (FULL) SEEK ADDRESS
000005              873     QWORD   EQU     SEKAD+1         (FULL) FAKE IOC QUEUE WORD
000006              874     URET    EQU     QWORD+1         (UPPER) USER RETURN ADDRESS
                    875                     CALLING         PARAMETERS
000006              876     ADEXT   EQU     URET            (LOWER) ADDRESS EXTENSION (BITS 0-5 OF 24 BIT ADD.)     [05NOV77]
000007              877     MODE    EQU     URET+1          (UPPER) MODE OF OPERATION
000007              878     MODEL   EQU     MODE            (LOWER)                                                 [09DEC79]
000010              879     DAC     EQU     MODE+1          (FULL) DEVICE ADDRESS CODE
000011              880     QUEWD   EQU     DAC+1           (FULL) PACKED VERSION OF QWORD ON RETURN
000012              881     DCWWD   EQU     QUEWD+1         (FULL) DCW RESIDUE
000013              882     SIDCW   EQU     DCWWD+1         (FULL) SEEK IDCW WHEN NEEDED
000014              883     SKDCW   EQU     SIDCW+1         (FULL) SEEK DCW WHEN NEEDED
000015              884     IDCW    EQU     SKDCW+1         (FULL) IOM COMMAND
000016              885     DCW     EQU     IDCW+1          (FULL) BEGINNING OF DCW LIST
                    886     *
                    887             UPPER   DEV,PUB,URET,MODE                                                       [09DEC79]
                    888             LOWER   TYPE,PUBL,SPRET,MODEL,ADEXT                                             [09DEC79]
                    889     *                                                                                       [09DEC79]
                    890     *       IOM CHANNEL DEFINITIONS:                                        16AUG74
                    891     *                                                                       16AUG74
                    892             HEAD    0                                                       16AUG74
000040              893     NCHAN   EQU     32              TOTAL NUMBER OF CHANNELS                16AUG74
000010              894     FPCHN   EQU     8               FIRST PAYLOAD CHANNEL IS 8              16AUG74
                    895             HEAD    X                                                       16AUG74
000004              896     FAUCH   EQU     1*4             FAULT CHANNEL * 4                       16AUG74
000010              897     CONCH   EQU     2*4             CONNECT CHANNEL * 4                     16AUG74
000014              898     SNACH   EQU     3*4             SNAPSHOT CHANNEL * 4                    16AUG74
000020              899     WRACH   EQU     4*4             WRAP AROUND CHANNEL * 4                 16AUG74
000024              900     BTCH    EQU     5*4             BOOTLOAD CHANNEL * 4                    16AUG74
000030              901     SPECH   EQU     6*4             SPECIAL STATUS CHANNEL * 4              16AUG74
000034              902     SCRCH   EQU     7*4             SCRATCH PAD CHANNEL * 4                 16AUG74
000000              903     LPW     EQU     0               RELATIVE LOCATION OF LPW IN MBX         16AUG74
000001              904     LPWX    EQU     1               RELATIVE LOCATION OF LPWX IN MBX        16AUG74
000002              905     SCW     EQU     2               RELATIVE LOCATION OF SCW IN MBX         16AUG74
000003              906     DCW     EQU     3               RELATIVE LOCATION OF DCW IN MBX         16AUG74
```

X DEFINITIONS --PHYSICAL I/O DEFINITIONS

```
                    907             EJECT                                                                   [09DEC79]
                    908     *
                    909     *
                    910     *       IOM COMMUNICATIONS AREA
                    911     *
                    912     *                                                                               [09DEC79]
                    913             HEAD    X                                                               [09DEC79]
                    914     *                                                                               [09DEC79]
            001400  915     MBX     EQU     V$MBX           128 WORDS -- 4 WORD MAILBOXES FOR 32 CHANNELS
                    916
                    917
            001340  918     IMW     EQU     V$IMW           32 WORDS -- INTERRUPT WORDS - ONE PER CHANNEL
                    919
                    920
            002340  921     SISTK   EQU     V$SISTK         32 WORDS -- SYSTEM INTERRUPT STATUS STACK
                    922     *
                    923     *
            001412  924     PCWA    EQU     MBX+CONCH+2     WHERE WE PUT PCWA FOR I/O
            001413  925     PCWB    EQU     PCWA+1          SAME FOR PCWB
                    926     *
            000001  927     BPCWB   EQU     1               BOOTLOAD PCWB CONTAINING CHANNEL NO AND IOM PORT
                    928     *
                    929     *                                                                       16AUG74
                    930     *       PROTECTED CORE LOCATIONS:                                       16AUG74
                    931     *                                                                       16AUG74
                    932             HEAD    Z                                                       16AUG74
                    933     *
                    934     *       BITS FOR I$QUEWD                                                        [05NOV77]
                    935     *                                                                               [05NOV77]
                    936             HEAD    B                                                               [05NOV77]
                    937     *                                                                               [05NOV77]
            000100  938     IORET   BOOL    000100          RET CODE FIELD                                  [05NOV77]
```

B DEFINITIONS --PHYSICAL I/O DEFINITIONS

```
                    939             EJECT                                                                   [09DEC79]
                    940     *                                                                               [05NOV77]
                    941     *       I$MODE--MODE OF OPERATION
                    942     *
                    943     *       THIS IS A CODE SPECIFYING THE TYPE OF OPERATION.  IT MAY HAVE
                    944     *       THE FOLLOWING VALUES
                    945     *
                    946             HEAD    I
                    947     *
          070000    948     MDRS    BOOL    070000          RESET MPC,CONSOLE                               [01SEP79]
          100000    949     MDAR    BOOL    100000          AWAIT READY
          110000    950     MDAS    BOOL    110000          AWAIT SPECIAL INTERRUPT
          140000    951     MDRV    BOOL    140000          USE NORMAL ERROR RECOVERY
          150000    952     MDNR    BOOL    150000          SUPRESS ALL ERROR RECOVERY
          200000    953     MDSB    BOOL    200000          SET BINARY (OR H716 NORMAL MODE)
          210000    954     MDSD    BOOL    210000          SET DECIMAL (OR H716 DUMP MODE)
          220000    955     MDSE    BOOL    220000          SET ASCII <--> EBCDIC MODE FOR 9 TRACK TAPES    [04JUL77]
          231000    956     MDD1    BOOL    231000          SET 200 BPI                                     [04JUL77]
          232000    957     MDD2    BOOL    232000          SET 556 BPI                                     [04JUL77]
          233000    958     MDD3    BOOL    233000          SET 800 BPI                                     [04JUL77]
          234000    959     MDD4    BOOL    234000          SET 1600 BPI                                    [04JUL77]
          235000    960     MDD5    BOOL    235000          SET 6250 BPI                                    [01MAY79]
          240000    961     MDSH    BOOL    240000          SET DEFAULT HIGH DENSITY                        [04JUL77]
          250000    962     MDSL    BOOL    250000          SET DEFAULT LOW DENSITY                         [04JUL77]
          260000    963     MDSA    BOOL    260000          SET ASCII
          270000    964     MDSP    BOOL    270000          SET FILE PROTECT
          300000    965     MDFR    BOOL    300000          FORWARD SPACE RECORD
          310000    966     MDBR    BOOL    310000          BACKSPACE RECORD
          320000    967     MDFF    BOOL    320000          FORWARD SPACE FILE
          330000    968     MDBF    BOOL    330000          BACKSPACE FILE
          340000    969     MDER    BOOL    340000          WRITE BLANK TAPE/ CONSOLE ALARM
          351000    970     MDDSE   BOOL    351000          DATA SECURITY ERASE                             [01MAY79]
          350000    971     MDEF    BOOL    350000          WRITE END-OF-FILE
          360000    972     MDWO    BOOL    360000        XX WRITE ONE CHAR RECORD XX
          370000    973     MDRW    BOOL    370000          REWIND
          371000    974     MDRU    BOOL    371000          REWIND AND UNLOAD
          400000    975     MDRD    BOOL    400000          READ                                            [04JUL77]
          410000    976     MDMR    BOOL    410000          MULTI RECORD READ                               [04JUL77]
          420000    977     MDRH    BOOL    420000          READ TRACK HEADER (DISK)                        [01MAY79]
          430000    978     MDFT0   BOOL    430000          FORMAT TRACK (DISK) 00                          [01MAY79]
          431000    979     MDFT1   BOOL    431000                              01                          [01MAY79]
          432000    980     MDFT2   BOOL    432000                              10                          [01MAY79]
          433000    981     MDFT3   BOOL    433000                              11                          [01MAY79]
          600000    982     MDWR    BOOL    600000          WRITE                                           [04JUL77]
          610000    983     MDMW    BOOL    610000          MULTI RECORD WRITE                              [04JUL77]
          620000    984     MDWI    BOOL    620000          WRITE IMAGE (PRINTER)                           [04JUL77]
          630000    985     MDWV    BOOL    630000          WRITE VFC (PRINTER)                             [04JUL77]
          640000    986     MDLC    BOOL    640000          LOAD CONTROL STORE (MPC)                        [04JUL77]
          650000    987     MDLM    BOOL    650000          LOAD MAIN MEMORY (MPC)                          [04JUL77]
          660000    988     MDLP    BOOL    660000          LOAD PERSONALITY (MPC)                          [04JUL77]
          670000    989     MDDS    BOOL    670000          READ DETAIL STATUS                              [04JUL77]
          700000    990     MDDG    BOOL    700000      XXXXX DIAGNOSTIC--SPECIAL FORMAT XXXXX
```

```
I                       DEFINITIONS --PHYSICAL I/O DEFINITIONS

                  991   *
                  992   *       INCLUDED HERE ARE SYMBOLS TO REFERENCE THE CATALOG TRACKS LIMITS
                  993   *       TABLE ENTRIES.
                  994   *
000000            995   CNLOW   BOOL    0               OFFSET FOR LOWER ADDRESS LIMIT
000001            996   CNUPR   BOOL    1               OFFSET FOR CATALOG TRACKS UPPER LIMIT
                  997   *                                 RREG
                  998
                  999
                 1000           HEAD    0
                 1001
002300           1002   CFILE   EQU     V$CFILE         BEGINNING OF CORE FILE POINTERS
                 1003
                 1004           HEAD    Z
                 1005
000100           1006   LLEN    EQU     32*2            LENGTH OF DCW LIST BLOCK FOR DISK BOOT
                 1007   *
                 1008   *       QUEUE SETS UP A QUEUE DESCRIPTER VECTOR
                 1009   *       WHICH HAS THE FOLLOWING FORM.
                 1010   *
                 1011   *       WORD            CONTENT
                 1012   *       ----            -------
                 1013   *       -1      Q$NPRI(0-17)    Q$BUSY(18-35)
                 1014   *       0       Q$START(0-17)   0(30-35)
                 1015   *       1       Q$INDEX(0-17)   TAG(30-35)
                 1016   *       ...     ...             ...
                 1017   *       NPRI    ...             ...
                 1018   *
                 1019   *
                 1020   *
                 1021           HEAD    Q
                 1022   *
777777           1023   NPRI    EQU     -1              UPPER HALF-NUMBER OF PRIORITY LEVEL
777777           1024   BUSY    EQU     -1              LOWER HALF--POINTER TO ACTIVE BLOCK
000001           1025   INDEX   EQU     +1              START OF PRIORITY INDEXES
000000           1026   START   EQU     0               FIRST ENTRY IN QUEUE
                 1027   *
                 1028           UPPER   NPRI                                                    [09DEC79]
                 1029           LOWER   BUSY                                                    [09DEC79]
                 1030   *                                                                       [09DEC79]
                 1031   *       TIMER QUEUE LIST ELEMENT DEFINITION
                 1032   *
                 1033   *
                 1034           HEAD    X
                 1035   *
000001           1036   TIM     EQU     1               PLACE WHERE TIME IS STORED
                 1037   *
                 1038   *       *  CSS  **
                 1039   *
                 1040   *         SCHEDULING FACTOR COMMONLY REFERRED TO ENTITIES     CSS
                 1041   *
000017           1042   WORP    EQU     15              PRIORITY NUMBER WHERE WORST IS DEFINED
```

X DEFINITIONS --PHYSICAL I/O DEFINITIONS

```
000004  1043  SPDEF   EQU     4               DEFAULT SCHEDULING PRIORITY (UNITY)
              1044  *       *  CSS  **
              1045  *
```

X DEFINITIONS --PHYSICAL DEVICE TYPES

```
                    1046            TTLSS   PHYSICAL DEVICE TYPES                                   [21APR77]
                    1047    *                                                                       [21APR77]
                    1048    *                                                                       [21APR77]
                    1049    *       PHYSICAL DEVICE TYPES (THE VALUES IN U$PTYPE)                   [21APR77]
                    1050    *                                                                       [21APR77]
                    1051            HEAD    U                                                       [21APR77]
                    1052                                                                            [21APR77]
000001              1053    D170    EQU     1               DSS170(2314) AND DSS180                 [21APR77]
000002              1054    M201    EQU     2               MDU201 DRUM                             [21APR77]
000003              1055    D167    EQU     3               DSS167 DISK                             [21APR77]
000004              1056            EQU     4               SPARE                                   [21APR77]
000005              1057            EQU     5               SPARE                                   [21APR77]
000006              1058    DUAL    EQU     6               SPLIT DEVICE (DUAL DRUMS)               [21APR77]
000007              1059    TAP7    EQU     7               7 TRACK MAG TAPE                        [21APR77]
000010              1060    TAP9    EQU     8               9 TRACK MAG TAPE                        [21APR77]
000011              1061    CON     EQU     9               CONSOLE                                 [21APR77]
000012              1062    RDR     EQU     10              CARD READER                             [21APR77]
000013              1063    PN10    EQU     11              100 CPM CARD PUNCH                      [21APR77]
000014              1064    P201    EQU     12              PRT201 PRINTER                          [21APR77]
000015              1065    DN30    EQU     13              DATANET 30                              [21APR77]
000016              1066    PN30    EQU     14              300 CPM CARD PUNCH                      [21APR77]
000017              1067    D191    EQU     15              DSS191 AND MSU400 DISK                  [21APR77]
000020              1068    H716    EQU     16              HISI 716                                [21APR77]
000021              1069    P301    EQU     17              PRT301 PRINTER                          [21APR77]
000022              1070    P400    EQU     18              PRT400 AND PRU1200/1600 PRINTER         [21APR77]
000023              1071    MPC     EQU     19              MPC                                     [21APR77]
000024              1072    M451    EQU     20              MSU451 DISK                             [21APR77]
000025              1073    L6      EQU     21              HISI LEVEL 6 FEP                        [09DEC79]
                    1074                                                                            [09DEC79]
000026              1075    PTYMX   EQU     22              MAX+1 LEGAL PHYSICAL DEVICE TYPES       [09DEC79]
                    1076    *                                                                       [21APR77]
                    1077    *       PERIPHERAL ALLOCATION TYPES (THE VALUES IN D$ATYPE)             [21APR77]
                    1078    *                                                                       [21APR77]
                    1079            HEAD    D                                                       [21APR77]
                    1080                                                                            [21APR77]
000001              1081    M201    EQU     1               MDU201 DRUM                             [21APR77]
000002              1082    D170F   EQU     2               DSS170(2314) AND DSS180 FILE TRACKS     [21APR77]
000003              1083    D170C   EQU     3               DSS170(2314) AND DSS180 CATALOG TRACKS  [21APR77]
000004              1084    D167    EQU     4               DSS167 DISK                             [21APR77]
000005              1085            EQU     5               SPARE                                   [21APR77]
000006              1086    DUAL    EQU     6               SPLIT DEVICE (DUAL DRUMS)               [21APR77]
000007              1087    D170    EQU     7               DSS170(2314) AND DSS180 ENTIRE PACK.    [21APR77]
000010              1088    D191    EQU     8               DSS191 AND MSU400 DISK, ENTIRE PACK     [21APR77]
000011              1089    D191C   EQU     9               DSS191 AND MSU400 DISK, CATALOG TRACKS  [21APR77]
000012              1090    D191F   EQU     10              DSS191 AND MSU400 DISK, FILE TRACKS     [21APR77]
000013              1091    M451    EQU     11              MSU451 DISK, ENTIRE PACK                [21APR77]
000014              1092    M451C   EQU     12              MSU451 DISK, CATALOG TRACKS             [21APR77]
000015              1093    M451F   EQU     13              MSU451 DISK, FILE TRACKS                [21APR77]
                    1094                                                                            [21APR77]
000016              1095    ATYMX   EQU     14              MAX + 1 LEGAL ALLOCATION TYPE           [21APR77]
```

D GENERAL PURPOSE MACRO DEFINITIONS

```
1096            TTLS    GENERAL PURPOSE MACRO DEFINITIONS
1097    *
1098    *
1099    *       MACRO TO CALL ALLOCATION ROUTINE TO ALLOCATE N WORDS OF A GIVEN PREF.
1100    *       CALL WITH T POINTING TO A CONTROL LIST ELEMENT, ARGUMENT #1 IS N, #2
1101    *       IS OK RETURN, #3 IS ERROR RETURN FOR OUT OF STORAGE, #4 IS PREFERENCE.
1102    *       IF ARGUMENT #5 IS 'GET' THEN THE MACRO WILL GENERATE A GET OF
1103    *       THE NECESSARY CONTROL LIST ELEMENT.
1104    *
1105    ALC     MACRO   N,OK,NOK,PREF,(GET)
1106            IFE     '#5','GET',1
1107            GETD    8               GET A CONTROL LIST ELEMENT FOR ALLOCATION
1108            LDA     #1              GET LENGTH TO ALLOCATE
1109            STA     K$NDA,T         SAVE IN LIST ELEMENT
1110            LDA     #2,DU           OK RETURN
1111            ADA     #3,DL           ERROR RETURN
1112            STA     K$RET,T         SAVE RETURN FROM ALLOCATION
1113            LDA     #4              GET PREFERENCE
1114            STA     K$TYPE,T        SAVE FOR ALLOCATION
1115            TRA     K$ALC           ALLOCATE A BLOCK                    [09DEC79]
1116            ENDM    ALC
1117    *
```

D LIST ELEMENT MACRO DEFINITIONS

```
1118            TTLS    LIST ELEMENT MACRO DEFINITIONS
1119    *
1120    *
1121    *
1122    PROTO   MACRO   POINTER          COPY PROTOTYPE LIST ELEMENT
1123            EAX     X,#1             POINT TO PROTOTYPE
1124            TSX0    E$PROTO          CALL SUBROUTINE TO COPY
1125            ENDM    PROTO
1126    *
1127    *
1128    *
1129    INVERT  MACRO                    INVERT TOP TWO ELEMENTS OF LIST
1130            TSX0    E$INVT           CALL SUBROUTINE
1131            ENDM    INVERT
```

D MULTI-PROCESSOR CODE GENERATION MACROS

```
1132            TTLS    MULTI-PROCESSOR CODE GENERATION MACROS
1133    *
1134    *           A   P   R   O   C
1135    *
1136    *       THIS MACRO REPEATS THE MACRO ARGUMENT ONCE FOR EACH
1137    *       PROCESSOR NUMBER IN THE SYSTEM, UP TO A MAXIMUM OF
1138    *       EIGHT PROCESSORS.
1139    *
1140    APROC   MACRO   MACRO-NAME
1141            PMC     SAVE,OFF
1142            CRSM    SAVE,OFF
1143            PROS    (0,1,2,3,4,5,6,7),(#1),(#2)
1144            CRSM    RESTORE
1145            ENDM    APROC
1146    *
1147    *           S   P   R   O   C
1148    *
1149    *       THIS MACRO REPEATS THE MACRO ARGUMENT ONCE FOR EACH NON-
1150    *       CONTROL PROCESSOR IN THE SYSTEM, UP TO A MAXIMUM
1151    *       OF EIGHT PROCESSORS.
1152    *
1153    SPROC   MACRO   MACRO-NAME
1154            PMC     SAVE,OFF
1155            CRSM    SAVE,OFF
1156            PROS    (1,2,3,4,5,6,7),(#1),(#2)
1157            CRSM    RESTORE
1158            ENDM    SPROC
1159    *
1160    *       PROS IS A SUBSIDIARY MACRO CALLED BY APROC AND SPROC
1161    *
1162    PROS    MACRO   LIST,NAME
1163            PMC     RESTORE
1164            IDRP    #1
1165            IFL     #1,$NPROS,4                                    16AUG74
1166            IFE     '#3',''
1167            #2      #1
1168            INE     '#3',''
1169            #2      #3#1
1170            IDRP
1171            ENDM    PROS
```

D

INTERRUPT CONTROL MACROS

```
1172              TTLS    INTERRUPT CONTROL MACROS
1173    *
1174    *
1175    *
1176    *                 MACROS TO MASK INTERRUPTS ON/OFF
1177    *
1178    DABL      MACRO
1179              RMCM    X$MEM           READ MASK FROM MEMORY CONTROLLER
1180              ANAQ    X$DABL          DISABLE SPEC-INIT-TERM-MARK          [09DEC79]
1181              SMCM    X$MEM           SET NEW MASK
1182              ENDM    DABL
1183    *
1184    ENABL     MACRO
1185              RMCM    X$MEM           READ MEMORY CONTROLLER MASK
1186              ORAQ    X$ENABL         ENABLE SPEC-TERM-MARK-CPAR           [09DEC79]
1187              SMCM    X$MEM           SET THE MASK
1188              ENDM    ENABL
```

D

```
                        BUG -- DESTROY REGISTERS

1189                    TTLS    BUG -- DESTROY REGISTERS
1190    *
1191    *
1192    *
1193    *       BUGBUG MUST BE DEFINED WITHIN THE SEGMENT WHICH USES IT. FOR
1194    *       DEBUGGING CLARITY, IT MUST BE INITIALIZED TO A DIFFERENT VALUE FOR
1195    *       EACH SEGMENT. HERE IS A LIST OF BUGBUG VALUES FOR THE SEGMENTS.
1196    *       IT SHOULD BE ADDED TO WHEN ANOTHER SEGMENT BUGS ITS REGISTERS.
1197    *
1198    *               MMES -- 750000(8)
1199    *               UTIL -- 750100(8)
1200    *               LOG  -- 750200(8)
1201    *               NEXT SEGMENT TO USE IT SHOULD SET IT TO 750300(8)
1202    *
1203    *
1204    *
1205    *       THIS MACRO GENERATES CODE THAT LOADS A DISTINCTIVE PATTERN
1206    *       INTO THE REGISTERS MENTIONED IN THE ARGUMENT LIST. THIS IS
1207    *       TO INSURE THAT REGISTERS ARE NOT ACCIDENTALLY CORRECT, SO
1208    *       THAT ERRORS MAY BE DETECTED AT AN EARLY PHASE OF THE DEBUGGING.
1209    *
1210    *       THE USAGE OF THIS MACRO IS BUG (A,Q,0,T,X3)
1211    *       WHERE THE REGISTERS IN THE LIST ARE THOSE WHICH WILL BE
1212    *       DESTROYED.  THE REGISTERS MENTIONED ARE ASSUMED TO BE INDEX
1213    *       REGISTERS UNLESS A OR Q OR AQ
1214    *
1215    *
1216    BUG     MACRO
1217            IFE     BUGBUG,0,1
1218            YOU     MUST            DEFINE BUGBUG FOR THIS SEGMENT. SEE BUG MACRO IN INSERT FILE
1219            INE     $DEBUG,0,5      SKIP IF NOT DEBUGGING
1220            PMC     SAVE,OFF
1221            IDRP    #1
1222            BUGA    #1              BUG ALL REGISTERS FROM CALL
1223            IDRP
1224            PMC     RESTORE
1225            ENDM    BUG
1226    BUGA    MACRO                   SUBSIDIARY MACRO TO BUG ONLY ONE REGISTER
1227            CRSM    SAVE,ON         CREATE SYMBOLS
1228            PMC     ON              LIST OUR USES
1229            IFE     '#1','A',A ASSEMBLE FOLLOWING BLOCK IFF WE ARE BUGGING R-A
1230            LDA     BUGBUG,DU
1231            ORA     BUGBUG,DL
1232            IFE     0,1,EOM IGNORE THE REST OF THIS
1233    A       MARK
1234    *
1235            IFE     '#1','Q',Q ASSEMBLE NEXT BLOCK IFF BUGGING R-Q
1236            LDQ     BUGBUG,DU
1237            ORQ     BUGBUG,DL
1238            IFE     0,1,EOM SKIP TILL END OF MACRO
1239    Q       MARK
1240    *
```

D

```
                        BUG -- DESTROY REGISTERS

1241                    IFE     '#1','AQ',AQ NEXT BLOCK IFF BUGGING R-AQ
1242                    LDA     BUGBUG,DU
1243                    ORA     BUGBUG,DL
1244                    LDQ     BUGBUG,DU
1245                    ORQ     BUGBUG,DL
1246                    IFE     0,1,EOM SKIP TILL END
1247    AQ              MARK
1248    *
1249                    LDX     #1,BUGBUG,DU
1250                    IFE     0,1,EOM                                    [04JUL77]
1251    EOM             MARK
1252                    PMC     OFF             KEEP THE LISTING NEET
1253    BUGBUG  SET     BUGBUG+1        INCREMENT BUGGING CONSTANT
1254                    CRSM    RESTORE
1255                    ENDM    BUGA
```

D CKPT -- CHECKPOINT MACRO

```
1256            TTLS    CKPT -- CHECKPOINT MACRO
1257    *
1258    *
1259    *
1260    CKPT    MACRO                   FOLLOWED BY A NUMBER FROM 0-35
1261            INE     $DEBUG,0,1      SKIP IF NOT DEBUGGING
1262            XED     $CKPT
1263            ENDM    CKPT
```

```
D                       QUEUING MACROS

                1264            TTLS    QUEUING MACROS
                1265    *
                1266    *
                1267    *
                1268    *
                1269    *               Q  U  E  U  E
                1270    *
                1271    *       THE USAGE OF THIS MACRO IS QUEUE,QDV,PRIOR WHERE QDV IS THE
                1272    *       NAME TO BE ASSIGNED TO THE QUEUE, AND PRIOR IS AN OPTIONAL
                1273    *       NUMBER OF PRIORITY LEVELS TO BE ASSIGNED, SET TO 2 IF NOT SPECIFIED.
                1274    *
                1275    QUEUE   MACRO
                1276            CRSM    SAVE,OFF
                1277    QSET    SET     2               ASSUME 2-LEVEL QUEUE
                1278            INE     #2,''           UNLESS SPECIFIED OTHERWISE,IN
                1279    QSET    SET     #2              WHICH CASE USE THE SPECIFICATION
                1280            ZERO    QSET,0          INITIALLY NOT BUSY
                1281    #1      ARG     0               LAST ELEMENT POINTER
                1282            ARG     0               PRIORITY 1 INDEX
                1283            INE     QSET,1,2        CAN'T DUP 0 TIMES
                1284            DUP     1,QSET-1        DEVELOP REST OF QUEUE
                1285            ARG     *-1,N*          INDIRECTION
                1286            CRSM    RESTORE
                1287            ENDM    QUEUE
                1288    *
                1289    *               P    T    Q
                1290    *
                1291    *       PTQ LOADS THE ADDRESS OF THE LIST ELEMENT ON THE TOP OF THE
                1292    *       SPECIFIED QUEUE INTO THE SPECIFIED INDEX REGISTER.  ITS USAGE
                1293    *       IS PTQ  XR,QDV  WHERE XR IS THE REGISTER AND QDV IS THE ADDRESS
                1294    *       OF THE QUEUE-DESCRIPTOR  VECTOR.
                1295    *
                1296    PTQ     MACRO                   POINT TO QUEUE
                1297            LDX     #1,#2           LOAD POINTER TO LIST ELEMENT OR ZERO
                1298            ENDM    PTQ             ZERO INDICATOR SET IF EMPTY QUEUE
```

D

QUEUING MACROS

```
1299            EJECT
1300    *
1301    *
1302    *                       E    N    Q
1303    *
1304    *       ENQ PLACES A LIST ELEMENT INTO A QUEUE AT THE PROPER PRIORITY
1305    *       LEVEL.  ITS USAGE IS ENQ  XR,QDV,PRIOR  WHERE XR IS AN INDEX
1306    *       REGISTER WHICH POINTS TO THE LIST ELEMENT TO BE QUEUED,  QDV
1307    *       IS THE QUEUE-NAME, AND PRIOR IS THE OPTIONAL PRIORITY
1308    *       OF THE ENTRY (2 ASSUMED IF NOT SPECIFIED)
1309    *
1310    ENQ     MACRO                       ENQUEUE
1311            CRSM    SAVE,OFF
1312            INE     #1,X,1              IF LIST ELEMENT NOT ALREADY IN X
1313            EAX     X,0,#1              PUT IT THERE
1314            IFE     '#2','$MTASK',5     CHECK FOR $MTASK
1315            INE     '#3',1,2            CHECK FOR LOW PRIORITY TASK
1316            TSX0    Q$MTQL              QUEUE WITH LOW PRIORITY
1317            IFE     0,1
1318            TSX0    Q$MTQH              QUEUE WITH HIGH PRIORITY
1319            IFE     0,1,7               SKIP REST OF MACRO
1320            EAX     Y,#2                QUEUE-DESCRIPTOR VECTER
1321            INE     '#3','Z',4
1322    QSET    SET     2                   ASSUME NO PRIORITY SPECIFIED
1323            INE     #3,''               IF NOT BLANK
1324    QSET    SET     #3                  SET IT TO SPECIFIED LEVEL
1325            EAX     Z,QSET              PRIORITY
1326            TSX0    Q$ENQ               GO TO ENQUEUE CODE
1327            CRSM    RESTORE
1328            ENDM    ENQ
1329    *
1330    *
1331    *                       M    T    Q
1332    *
1333    *       MTQ QUEUES THE TASK POINTED TO BY REGISTER T ON THE
1334    *       MASTER TASK QUEUE.  IT'S FUNCTION IS IDENTICAL TO THE
1335    *       EXPANSION OF 'ENQ T,$MTASK', BUT REQUIRES LESS CODE
1336    *
1337    *
1338    MTQ     MACRO                       QUEUE A MASTER TASK
1339            TSX0    Q$MTQ               GO QUEUE THE TASK
1340            ENDM    MTQ
1341    *
1342    *                                       M T Q A
1343    *
1344    *       MTQA QUEUES THE TASK BLOCK POINTED TO BY REGISTER T
1345    *       WITH THE RESTART LOCATION IN AL  ON THE MASTER TASK QUEUE
1346    *
1347    MTQA    MACRO
1348            TSX0    Q$MTQA              CALL SUBROUTINE TO QUEUE TASK
1349            ENDM    MTQA
1350    *
```

D

QUEUING MACROS

```
1351    *
1352    *
1353    *               D    E    Q
1354    *
1355    *       DEQ REMOVES THE TOP LIST ELEMENT FROM A QUEUE
1356    *       USAGE IS DEQ  XR,QDV  WHERE XR IS REGISTER WHICH WILL BE LOADED
1357    *       WITH THE ADDRESS OF THE TOP ELEMENT OF THE QUEUE.  QDV IS THE
1358    *       QUEUE-NAME.  THE ZERO INDICATOR IS SET ON IF THERE
1359    *       WAS NO BLOCK ON THE QUEUE.
1360    *
1361    DEQ     MACRO                   DEQUEUE
1362            EAX     X,#2            LOAD QUEUE NAME
1363            TSX0    Q$DEQ           EXECUTE NECESSARY CODE
1364            EAX     #1,0,Y          LOAD REGISTER WITH LIST ELEMENT ADDRESS
1365            ENDM    DEQ
1366    *
1367    *
1368    *                                                                    [170CT76]
1369    *       MTASK  --  CREATE A NEW MASTER TASK WITH OPTIONAL PARAMETER   [170CT76]
1370    *       THE CURRENT STACK OF LIST ELEMENTS WILL BE KEPT ONLY IF 'KEEP' [170CT76]
1371    *       IS SPECIFIED AS AN OPTIONAL THIRD PARAMETER                  [170CT76]
1372    *                                                                    [170CT76]
1373    MTASK   MACRO   LOC,PARAMETER,KEEP                                   [170CT76]
1374            IFE     '#2',',2                                             [170CT76]
1375            GETD    1,NBUG                                               [170CT76]
1376            IFE     0,1,9                                                [170CT76]
1377            GETD    2,NBUG                                               [170CT76]
1378            IFE     '#2','ZERO',2                                        [170CT76]
1379            STZ     1,T             PARAMETER IS ZERO                    [170CT76]
1380            IFE     0,1,5                                                [170CT76]
1381            IFE     '#2','STC1',2                                        [170CT76]
1382            STC1    1,T             PARAMETER IS IC/IR (NON-ZERO)        [170CT76]
1383            IFE     0,1,2                                                [170CT76]
1384            LDA     #2              PARAMETER IS #2                      [170CT76]
1385            STA     1,T             SAVE IT                              [170CT76]
1386            LDA     #1,DL           RESTART ADDRESS                      [170CT76]
1387            MTQA                    QUEUE TASK TO START AT #1            [170CT76]
1388            IFE     '#3','KEEP',2                                        [170CT76]
1389            LDX     T,T$LINK,T      POP BACK LIST ELEMENT                [170CT76]
1390            IFE     0,1,1                                                [170CT76]
1391            LDX     T,0,DU          SPPML                                [170CT76]
1392            ENDM    MTASK                                                [170CT76]
```

D

LIST ELEMENT ALLOCATION MACROS

```
1393            TTLS    LIST ELEMENT ALLOCATION MACROS
1394    *
1395    *
1396    *
1397    *       THE FOLLOWING MACROS MANIPULATE LIST ELEMENTS
1398    *       WHEN ENTERED XT POINTS TO WD 0 OF THE TOP ELEMENT OF A LINKED
1399    *       LIST (IF THE LIST IS EMPTY XT=0) WORD -1 OF A LIST ELEMENT
1400    *       CONTAINS  UPPER  A BACKWARD POINTER TO WD 0 OF THE PREVIOUS LIST
1401    *                 LOWER  THE LENGTH OF THIS ELEMENT NOT COUNTING THE LINK
1402    *       ALL MACROS POP AND PUSH XT AS APPROPRIATE
1403    *
1404    *       THE GET MACRO RETURNS A LIST ELEMENT OR DIES TRYING.  CALLS ARE
1405    *       GET     (LEN,I)     LENGTH RIGHT JUSTIFIED IN LOCATION LEN
1406    *       GET     AL              LENGTH RIGHT JUSTIFIED IN AL
1407    *       GET     AU              LENGTH RIGHT JUSTIFIED IN AU
1408    *       GETD    LENGTH          EXPLICIT LENGTH
1409    *
1410    *       EACH OF THESE MACROS HAS AN OPTIONAL SECOND PARAMETER 'NBUG'          [170CT76]
1411    *       SPECIFYING THAT THE LIST ELEMENT SHOULD NOT BE BUGGED SINCE           [170CT76]
1412    *       THE CALLING ROUTINE IS ABOUT TO FILL IT COMPLETELY.                   [170CT76]
1413    *                                                                             [170CT76]
1414    GET     MACRO   (LENGTH,I/AU/AL),NBUG                                         [170CT76]
1415            INE     '#1','AU',3
1416            INE     '#1','AL',1
1417            LDA     #1
1418            ALS     18
1419            IFE     '#2','NBUG',2                                                 [170CT76]
1420            TSX0    A$GETNB         CALL TO ENTRY THAT WILL NOT BUG THE LIST ELEMENT [170CT76]
1421            IFE     0,1,1                                                         [170CT76]
1422            TSX0    A$GET
1423            ENDM    GET
1424    GETD    MACRO   LENGTH,NBUG                                                   [170CT76]
1425            LDA     #1,DU
1426            IFE     '#2','NBUG',2                                                 [170CT76]
1427            TSX0    A$GETNB         CALL TO ENTRY THAT WILL NOT BUG THE LIST ELEMENT [170CT76]
1428            IFE     0,1,1                                                         [170CT76]
1429            TSX0    A$GET
1430            ENDM    GETD
1431    *
1432    *       THE GETB MACRO RETURNS TO OK IF SUCCESSFUL AND TO NOT OK IF NOT.
1433    *       IF EITHER OF THESE FIELDS IS NULL, *+1 IS ASSUMED FOR THAT FIELD
1434    *       LENGTH IS AS IN GET
1435    *                                                                             [05NOV77]
1436    GETB    MACRO   (LENGTH,I/AL/AU),(OK),(NOT  OK),(CALL BY GETBQ)               [05NOV77]
1437            INE     '#1','AU',3
1438            INE     '#1','AL',1
1439            LDA     #1
1440            ALS     18
1441            TSX0    A$BUF                                                         [05NOV77]
1442            INE     '#4','*',8      CHECK FOR CALL FROM GETBQ                     [05NOV77]
1443            IFE     '#2','',1                                                     [05NOV77]
1444            TRA     #3              UNAVAILABLE                                   [01MAY79]
```

D

LIST ELEMENT ALLOCATION MACROS

```
1445            INE     '#2','',5                                          [01MAY79]
1446            IFE     '#3','',1                                          [01MAY79]
1447            TRA     *+2             GOT IT                             [01MAY79]
1448            INE     '#3','',1                                          [01MAY79]
1449            TRA     #3              UNAVAILABLE                        [01MAY79]
1450            TRA     #2              GOT IT                             [01MAY79]
1451            ENDM    GETB
1452    *
1453    *       THE GETBQ MACRO RETURNS TO *+1 WITH THE LIST ELEMENT. IF ONE IS
1454    *       NOT IMMEDIATLY AVAILABLE THE REQUEST IS QUEUED AND HANDLED WHEN
1455    *       THE MEMORY BECOMES AVAILABLE.
1456    *       LENGTH IS AS IN GETB                                         [05NOV77]
1457    *                                                                    [05NOV77]
1458    GETBQ   MACRO   (LENGTH,I/AL/AU)                                     [05NOV77]
1459            CRSM    SAVE,OFF                                           [05NOV77]
1460            GETB    (#1),,,*        ATTEMPT TO GET A BUFFER            [05NOV77]
1461            TSX0    A$BUFQ          IF CURRENTLY UNAVAILABLE, QUEUE FOR ONE [05NOV77]
1462            CRSM    RESTORE
1463            ENDM    GETBQ
1464    *
1465    *       THE REL MACRO RELEASES THE BLOCK POINTED TO BY T AND POPS T.
1466    *
1467    REL     MACRO
1468            TSX0    A$REL
1469            ENDM    REL
1470    *
1471    *       THE SHRINK MACRO SHRINKS THE BLOCK POINTED TO BY XR TO THE
1472    *       LENGTH SPECIFIED
1473    *
1474    SHRINK  MACRO   (LENGTH,I/AU/AL),XR
1475            EAX     Y,0,#2          POINTER TO LIST ELEMENT
1476            INE     '#1','AU',3
1477            INE     '#1','AL',1
1478            LDA     #1
1479            ALS     18
1480            TSX0    A$SHRI
1481            ENDM    SHRINK
```

D

LIST ELEMENT ALLOCATION MACROS

```
1482            EJECT
1483    *
1484    *       THE EXPAND MACRO EXPANDS THE BLOCK POINTED TO BY THE LINK WORD
1485    *       OF THE BLOCK POINTED TO BY XR TO THE LENGTH SPECIFIED
1486    *
1487    EXPAND  MACRO   (LENGTH,I/AU/AL),XR
1488            INE     '#1','AU',3
1489            INE     '#1','AL',1
1490            LDA     #1
1491            ALS     18
1492            INE     'X','#2',1
1493            EAX     X,0,#2          POINTER TO POINTER
1494            TSX0    A$EXP
1495            ENDM    EXPAND
```

D

CONSOLE LOGGING MACROS

```
1496              TTLS    CONSOLE LOGGING MACROS
1497    *
1498    *
1499    *         LOG LOGS MESSAGES TO BOTH THE CONSOLE AND THE LOG FILE. CALL IT
1500    *         WITH UP TO TWELVE CHARACTERS OF TEXT AND UP TO FOUR ARGS TO BE
1501    *         PRINTED IN OCTAL.  THE LOG ROUTINE WILL APPEND THE TIME TO THE END.
1502    *
1503    LOG       MACRO   (TEXT),WORD1,...,WORD4
1504              CRSM    SAVE,OFF
1505              STZ     I$FLOG          DON'T INHIBIT DEVICE OUTPUT
1506              TSX     0,I$LOG         CAN BE CALLED FROM THE OUTSIDE WORLD
1507              BCI     2,#1            TEXT ARGUMENT                          [08AUG77]
1508              INE     'A#2','A'       ANY ARGUMENT HERE?
1509              ARG     #2              YES, POINT TO IT
1510              INE     'A#3','A'       SIMILARLY FOR THE REST
1511              ARG     #3
1512              INE     'A#4','A'
1513              ARG     #4
1514              INE     '#5',''
1515              ARG     #5
1516              CRSM    RESTORE
1517              ENDM    LOG
1518    *
1519    *         SAME AS ABOVE BUT NO LOGGING TO CONSOLE
1520    *
1521    LOGF      MACRO
1522              CRSM    SAVE,OFF
1523              STZ     I$FLOG                                                 [22JUN76]
1524              STC2    I$FLOG                                                 [22JUN76]
1525              TSX0    I$LOG
1526              BCI     2,#1
1527              INE     '#2',''
1528              ARG     #2
1529              INE     '#3',''
1530              ARG     #3
1531              INE     '#4',''
1532              ARG     #4
1533              INE     '#5',''
1534              ARG     #5
1535              CRSM    RESTORE
1536              ENDM    LOGF
1537    *
1538    ALARM     MACRO                   RING CONSOLE ALARM AFTER LOG
1539              INHIB   SAVE,OFF        UNINHIBIT
1540              NOP     91356,DU        SIGNAL ALARM WANTED
1541              INHIB   RESTORE         RESTORE INHIBIT
1542              ENDM    ALARM
```

D

CONSOLE LOGGING MACROS

```
1543            EJECT                                                   [22JUN76]
1544                                                                    [22JUN76]
1545    *       ORDER MACRO                                             [22JUN76]
1546                                                                    [22JUN76]
1547    *       THIS PRINTS A MESSAGE ON THE DEVICE ASSIGNED FOR        [22JUN76]
1548    *       COMMUNICATION WITH THE OPERATOR.                        [22JUN76]
1549                                                                    [22JUN76]
1550    *       ORDERS BEGIN WITH VERBS                                 [22JUN76]
1551                                                                    [22JUN76]
1552                                                                    [22JUN76]
1553    *            NOT TO BE CALLED FROM I/O INITIATION               [22JUN76]
1554                                                                    [22JUN76]
1555                                                                    [22JUN76]
1556    ORDER   MACRO   N,(TEXTSTRING)                                  [22JUN76]
1557            IFE     #1,0,1                                          [22JUN76]
1558            ARG     ERROR           IN ORDER MACRO                  [22JUN76]
1559            TSX     0,I$ORDER       GOTO SUBROUTINE                 [22JUN76]
1560            TRA     *+1+#1          TRANSFER AROUND TEXT            [22JUN76]
1561            BCI     #1,#2?????                                      [22JUN76]
1562            ENDM    ORDER                                           [22JUN76]
```

D COPY MACRO

```
1563              TTLS    COPY MACRO
1564    *
1565    *
1566    *
1567    *
1568    COPY      MACRO   NUMBER  OF WORDS TO TRANSFER,RESTART ADDRESS,[WORKING BLOCK]PAGES
1569              IFE     'A#3','A',2
1570              GETD    C$LISLN          GET A WORKING BLOCK
1571              IFE     1,2,2
1572              STX     T,T$LINK+#3      LINK USER WORK AREA PROPPERLY
1573              EAX     T,#3
1574              LCA     #1               GET MINUS NUMBER OF WORDS TO TRANSFER
1575              STA     C$STAT2,T        STORE IN LIST ELEMENT
1576              LDA     #2,DL            GET RESTART LOCATION ON COMPLETION OF COPY
1577              TRA     C$COPY           AND COPY
1578              ENDM    COPY
```

D COPY CONTROL LIST ELEMENT DEFINITION

```
                    1579            TTLS    COPY CONTROL LIST ELEMENT DEFINITION
                    1580    *
                    1581    *
                    1582    *
                    1583    *
                    1584            HEAD    C                   FOR COPY
                    1585
000001              1586    STAT2   EQU     1
000002              1587    TEMPN   EQU     2
000003              1588    OLDN    EQU     3
000004              1589    F2DA    EQU     4
000005              1590    BUFPT   EQU     5
000006              1591    STAT1   EQU     6
000007              1592    IODA    EQU     7
000010              1593    SKIP    EQU     8
000011              1594    CPYN    EQU     9
000012              1595    RX      EQU     10
000013              1596    RY      EQU     11
000014              1597    RZ      EQU     12
000015              1598    XFER    EQU     13
000016              1599    COM     EQU     14
000017              1600    OFST1   EQU     15
000020              1601    URET    EQU     16
000021              1602    RRET    EQU     17
000022              1603    LISLN   EQU     18              LENGTH OF LIST ELEMENT
```

C CATALOG CONTROL LIST ELEMENT DEFINITIONS

```
                    1604            TTLS    CATALOG CONTROL LIST ELEMENT DEFINITIONS                                [01MAY79]
                    1605    *
                    1606    *
                    1607    *           CATALOGS ARE MANIPULATED BY THE FOLLOWING MACROS AND SUBROUTINES.
                    1608    *       THESE SUBROUTINES CALL THE CATALOG ROUTINES IN THE NEXT SECTION TO QUEUE
                    1609    *       CATALOG REQUESTS AND TO PERFORM CATALOG IO.
                    1610    *       WITH THE EXCEPTION OF THE MACROS WSC, DUSE, ATACH, AND DTACH
                    1611    *       ALL CATALOG OPERATIONS ARE CONTROLLED BY A LIST ELEMENT POINTED TO
                    1612    *       BY XR - T.  THE CONTENTS OF THIS LIST ELEMENT ARE AS FOLLOWS:
                    1613    *
                    1614            HEAD    C                                                                       [01MAY79]
                    1615                                    WORD 0 CONTAINS Q$LINK / Q$RUN
             000001 1616    HDA     EQU     1               DEVICE ADDRESS OF CATALOG HEADER
             000002 1617    NAM     EQU     2               NAME OF FILE BEING SEARCHED FOR (TWO WORDS)
             000004 1618    PWORD   EQU     4               PASSWORD OF FILE (TWO WORDS)
             000006 1619    XR      EQU     6               USED FOR SAVING REGISTERS J, P, S, 0 (TWO WORDS)
             000010 1620    CATR    EQU     8               RETURN FROM CATALOG SUBROUTINE (UPPER)
             000010 1621    USEP    EQU     8               POINTER TO C$USE ENTRY (LOWER)
             000011 1622    PERM    EQU     9               USER PERMISSIONS FOR PROTECTION CHECKS (UPPER)
             000011 1623    FLAG    EQU     9               FLAGS CONTROLLING OPERATION (LOWER)
             000012 1624    CACC    EQU     10              ACCESS WORD
             000013 1625    DATES   EQU     11              CODED DLU AND DLM FOR INSC
             000014 1626    USAGE   EQU     12              USAGE COUNTER FOR INSC                                  [01MAY79]
             000015 1627    AQ1     EQU     13              USER'S REG A FOR SLAVE TRAP                             [01MAY79]
             000016 1628    AQ2     EQU     14              USER'S REG Q FOR SLAVE TRAPS                            [01MAY79]
             000017 1629    PASPT   EQU     15              POINTER TO PASSWORD LIST (UPPER)                        [01MAY79]
             000017 1630    PASLE   EQU     15              POINTER TO PASSWORD LIST ELEMENT IFF SEPARATE (LOWER)   [09DEC79]
             000020 1631    PWFLG   EQU     16              BIT CODED WORD TELLING WHICH LEVELS HAVE PWS            [01MAY79]
             000021 1632    NAMND   EQU     17              POINTER TO WORD BEYOND LAST WORD OF NAME IN TREE (UPPER) [01MAY79]
             000021 1633    NAMPT   EQU     NAMND           POINTER TO STARING (OR CURRENT IF IN CATS) NAME (LOWER) [01MAY79]
             000022 1634    CBITS   EQU     18              BITS FOR CCE (UPPER)                                    [01MAY79]
             000022 1635    MMENO   EQU     CBITS           MME NUMBER FOR SLAVE TRAPS (UPPER)                      [01MAY79]
             000022 1636    TACC    EQU     18              TERMBITS OF SLAVE TRAP (LOWER)                          [01MAY79]
             000023 1637    LELEN   EQU     19              LENGTH OF LIST ELEMENT HEAD                             [09DEC79]
                    1638    *                                                                                       [09DEC79]
                    1639            UPPER   CATR,PERM,PASPT,NAMND,CBITS,MMENO                                     [09DEC79]
                    1640            LOWER   USEP,FLAG,PASLE,NAMPT,TACC                                           [09DEC79]
                    1641    *
                    1642    *       BIT DEFINITIONS FOR C$FLAG
                    1643    *
             000001 1644    PWD     BOOL    1               USER HAS SUPPLIED A PASSWORD
             000002 1645    FILE    BOOL    2               AN FCB EXISTS BEHIND THE CONTROLLING LIST ELEMENT
             000004 1646    OWN     BOOL    4               USER HAS OWNER PERMISSION ON THE CATALOG
             000010 1647    QUOTA   BOOL    10              QUOTA CHECKS SHOULD BE PERFORMED ON THE FILE
             000020 1648    NDATE   BOOL    20              DO NOT CHECK CODED DATES IN FUSE SUBROUTINE
             000040 1649    NATCH   BOOL    40              GFCBC SHOULD NOT GET A C$USE LIST ENTRY
             000100 1650    CLI     BOOL    100             DO CLIMBING ACCESS FOR CATL AND CATS
             000200 1651    CLIM    BOOL    200             B$CLI NEEDED ON DESTINATION FILE OF CATL
             000400 1652    NTMSK   BOOL    400             FLAGS AN 'OLD' MME WHICH DOES NOT USE A TRAP MASK       [01MAY79]
                    1653                                    B$AP, B$RD, B$WT FLAG NEEDED ACCESSES ON CATALOG        [01MAY79]
             020000 1654    HELD    BOOL    20000           FILE IS HELD NON-DEALOCATABLE                           [01MAY79]
                    1655    *
```

C CATALOG CONTROL LIST ELEMENT DEFINITIONS

```
                    1656    *       BIT DEFINITIONS FOR C$CBITS
                    1657    *
            000001  1658    RNAM    BOOL    1               USER WISHES TO RENAME FILE
            000002  1659    RPWD    BOOL    2               USER WISHES TO REPASSWORD FILE
            000004  1660    RACC    BOOL    4               USER WISHES TO CHANGE ACCESSES
            000010  1661    RDAT    BOOL    10              USER WISHES TO CHANGE DLU/DLM
```

C CATALOG SUBROUTINES -- GENERAL MACROS

```
1662              TTLS    CATALOG SUBROUTINES -- GENERAL MACROS
1663    *
1664    *
1665    *         GENERAL MACROS
1666    *
1667    *
1668    *         THE FOLLOWING MACRO GETS A POINTER TO A FILE CONTROL BLOCK IN THE USER'S
1669    *         STATE VECTOR.  ARGUMENT #1 IS THE INDEX REGISTER CONTAINING THE FILE
1670    *         REFERENCE NUMBER OF THE DESIRED FCB.
1671    *
1672    FCBPNT    MACRO   INDEX
1673              LDX     Z,$FR,*#1       GET A RELATIVE POINTER TO THE FILE CONTROL BLOCK
1674              XED     E$FCBPT         CHECK VALIDITY AND MAKE ABSOLUTE
1675              ENDM    FCBPNT
1676    *                                                                          [170CT76]
1677    *         GFR  --  THIS MACRO AND THE CORRESPONDING SUBROUTINE FINDS A NEW [170CT76]
1678    *         FILE REFERENCE NUMBER AND RETURNS IT IN REGISTER 'X'.            [170CT76]
1679    *                                                                          [170CT76]
1680    GFR       MACRO                                                          [170CT76]
1681              TSX0    $GFR            CALL THE APPROPRIATE SUBROUTINE        [170CT76]
1682              ENDM    GFR                                                    [170CT76]
```

C QLOCK AND QNLOCK MACROS

```
1683            TTLS    QLOCK AND QNLOCK MACROS
1684    *                                                                [170CT76]
1685    *       MULTIPROCESSING QUEUE LOCK AND UNLOCK MACROS             [170CT76]
1686    *                                                                [170CT76]
1687    QLOCK   MACRO                                                    [170CT76]
1688            TSX0    X$QLOCK         LOWER THE SEMAPHORE              [170CT76]
1689            ENDM    QLOCK                                            [170CT76]
1690    *                                                                [170CT76]
1691    QNLOCK  MACRO                                                    [170CT76]
1692            XED     X$QNLCK         RAISE THE SEMAPHORE              [170CT76]
1693            ENDM    QNLOCK                                           [170CT76]
1694    *
1695    *
1696    *           THE WSC MACRO WRITES A SCRATCH CATALOG HEADER.  IT SHOULD BE
1697    *       CALLED WITH INDEX T POINTING TO A LIST ELEMENT CONTAINING THE
1698    *       PROTOTYPE FOR THE HEADER.  THE PREFERENCE DESIRED FOR THE CATALOG
1699    *       SHOULD BE IN LOCATION C$CDA OF THE LIST ELEMENT.  UPON RETURN THE
1700    *       LOCATION C$CDA WILL CONTAIN THE DEVICE ADDRESS OF THE CREATED
1701    *       HEADER.
1702    *           ARGUMENT #1 IS THE LOCATION FOR ERROR RETURNS (STATUS IS
1703    *       RETURNED IN THE A REGISTER), #2 IS A POINTER TO THE LENGTH WHICH
1704    *       SHOULD BE ALLOCATED FOR THE CATALOG.
1705    *
1706    *
1707    *
1708    WSC     MACRO                   WRITE SCRATCH CATALOG
1709            CRSM    SAVE,ON
1710            GETD    8               SET UP LIST ELEMENT FOR ALLOCATION
1711            LDX     X,T$LINK,T      GET POINTER TO CAT FCB IN LIST ELEMENT
1712            ALC     #2,#6,#3,(C$CDA,X)   ALLOCATE STORAGE FOR HEADER
1713    #3      REL                     RELEASE ALLOCATION CONTROL LIST ELEMENT
1714            LDA     B$NSTOR,DU      GET ERROR STATUS
1715            TRA     #1              ALLOCATION FAILURE, NO STORAGE, ERROR RETURN
1716    #6      LDA     K$NDA,T         DEVICE ADDRESS FOR NEW SCRATCH CATALOG HEADER
1717            LDX     X,T$LINK,T      GET POINTER TO CATALOG HEADER IMAGE
1718            STA     C$CDA,X         AND STORE IN IT THE NEW DEVICE ADDRESS
1719            REL                     RELEASE THE ALLOCATION CONTROL LIST ELEMENT
1720            GETD    I$DCW+1         SET UP LIST ELEMENT FOR IO
1721            LDA     I$MDWR,DU       GET WRITE COMMAND
1722            STA     I$MODE,T        AND STORE IN IO LIST ELEMENT
1723            LDX     X,T$LINK,T      GET POINTER TO CATALOG HEADER IMAGE
1724            LDA     C$CDA,X         AND GET DA FOR IO OPERATION
1725            STA     I$DAC,T         AND STORE IN IO LIST ELEMENT
1726            STX     X,I$DCW,T       STORE STARTING ADDRESS OF IO IN DCW
1727            EAQ     C$HEADL         LENGTH OF CATALOG HEADER
1728            STZ     C$CKSUM,X       INITIALIZE CHECKSUM FIELD
1729            TSX     Y,X$CKSUM       COMPUTE HEADER CHECKSUM
1730            LDX     X,T$LINK,T      POINT TO PROTOTYPE HEADER AGAIN
1731            STA     C$CKSUM,X       SET CHECKSUM IN HEADER
1732            LDX     X,C$HEADL+1,DU  GET NUMBER OF WORDS TO TRANSFER
1733            SXL     X,I$DCW,T       AND STORE IN DCW
1734            LDX0    0,DU            LOAD ADDRESS EXTENSION FOR CAT HEADER   [05NOV77]
```

C QLOCK AND QNLOCK MACROS

```
1735           SXLO    I$ADEXT,T       LIST ELEMENTS ARE IN FIRST 256K                    [05NOV77]
1736           TSX0    I$IO            DO IO OPERATION
1737           LDA     I$QUEWD,T       I/O COMPLETE, GET STATUS OF OPERATION
1738           ANA     =07700,DL        GET  RET  BITS, WHICH GIVE STATUS
1739           TZE     #5              TRANSFER IF IO OPERATION OK
1740    *
1741    *              ERROR IN IO OPERATION, SO DEALLOCATE PREVIOUSLY ALLOCATED SPACE
1742    *
1743           REL                     RELEASE IO LIST ELEMENT
1744           LDA     B$RERR,DU       GET ERROR STATUS
1745           TRA     #1              AND GIVE ERROR RETURN
1746    #5     REL                     IO IS OK, DONE, RELEASE IO LIST ELEMENT
1747           CRSM    RESTORE
1748           ENDM    WSC
```

C

```
                        CATALOG OPERATIONS MACROS

1749                    TTLS    CATALOG OPERATIONS MACROS                    [01MAY79]
1750    *
1751    *
1752    *       CATC IS A MACRO TO SIEZE THE CATALOG ROUTINES
1753    *       IT GETS A CATALOG CONTROL LIST ELEMENT AND THEN RETURNS
1754    *       WITHOUT READING IN A CATALOG
1755    *       AS USUAL, OPERATIONS STATARTING WITH CATC MUST TERMINATE
1756    *       WITH A RELC MACRO
1757    *
1758    *
1759    CATC    MACRO   (NO             ARGUMENTS)
1760            GETD    C$CACC+1        GET A CATALOG CONTROL LIST ELEMENT
1761            STZ     Q$RUN,T         NO TASK TO DO
1762            TSX     X,C$CAT         GO QUEUE THE OPERATION
1763            ENDM    CATC
1764    *
1765    *       GFDA MACRO CALLS THE GFDA SUBROUTINE TO GET A FILE CONTROL
1766    *       BLOCK FOR THE CURRENT ENTRY WITH JUST THE DA'S
1767    *
1768    GFDA    MACRO   (NO ARGUMENTS)                                       [01MAY79]
1769            TSXO    C$GFDA          USE THE SUBROUTINE
1770            ENDM    GFDA
```

C CATALOG OPERATIONS MACROS

```
1771            EJECT
1772    *
1773    *
1774    *           CATH IS A MACRO TO READ IN THE FIRST SEGMENT OF A CATALOG.
1775    *       ARGUMENT #1 FOR THE MACRO IS A POINTER TO THE DEVICE ADDRESS.
1776    *       POSSIBLE STATUS RETURNS IN C$CSTAT ARE
1777    *            B$OK       OPERATION SUCCESSFUL
1778    *            B$RERR     IO ERROR IN READING IN CATALOG
1779    *            B$UERR     INFORMATION IN THE CATALOG HAS BEEN DESTROYED
1780    *
1781    *
1782    CATH    MACRO   DAPTR,'TRAP'OR'' TRAP MEANS SET SLAVE TRAP STUFF
1783            CRSM    SAVE,OFF
1784            IFE     #2,'TRAP',2
1785            GETD    C$LELEN                                                 [09DEC79]
1786            IFE     0,1,1
1787            GETD    C$USAGE+1       GET A CONTROLLING LIST ELEMENT
1788            LDA     C$CRET,DL       GET ADDRESS OF RETURN ROUTINE
1789            STA     Q$RUN,T         SAVE FOR TRANSFER ONCE HEADER IS READ IN
1790            LDA     #1              GET DEVICE ADDRESS OF CATALOG
1791            STA     C$HDA,T         SAVE IN CONTROL BLOCK
1792            IFE     #2,'TRAP',11
1793            LDQ     S$REG+4,S                                               [29JAN77]
1794            STQ     C$AQ1,T                                                 [29JAN77]
1795            LDQ     S$REG+5,S                                               [29JAN77]
1796            STQ     C$AQ2,T                                                 [29JAN77]
1797            LDX     X,S$IC,S        GET POINTER TO MME+1
1798            ADLX    X,$BASE         MAKE ABSOLUTE                           [08AUG77]
1799            MLDA    -1,X            GET MME NUMBER                          [08AUG77]
1800            ANA     -1,DU           ONLY
1801            ALS     9               SANS MME ID
1802            EAX     X,0,AU          PUT INTO X
1803            STX     X,C$MMENO,T     SAVE FOR SLAVE TRAP
1804            TSX     X,C$CAT         SET RETURN IN XR-X AND ENTER CATALOG ROUTINE
1805            CRSM    RESTORE
1806            ENDM    CATH
```

C

```
                         CATALOG OPERATIONS MACROS

1807                     EJECT                                                  [01MAY79]
1808     *                                                                      [01MAY79]
1809     *                                                                      [01MAY79]
1810     *       CATL AND TCATL ARE MACROS FOR CALLING THE CATALOG LOOKUP ROUTINE [01MAY79]
1811     *       CATL IS USED FOR 'OLD' OR 'FETCHING' AND TCATL IS USED FOR 'TALLY' OPS. [01MAY79]
1812     *                                                                      [01MAY79]
1813     CATL    MACRO                                                          [01MAY79]
1814             TSXO    C$CTENT                                                [01MAY79]
1815             ENDM    CATL                                                   [01MAY79]
1816     *                                                                      [01MAY79]
1817     *                                                                      [01MAY79]
1818     TCATL   MACRO                                                          [01MAY79]
1819             TSXO    C$TLENT                                                [01MAY79]
1820             ENDM    TCATL                                                  [01MAY79]
```

C CATALOG OPERATIONS MACROS

```
1821            EJECT
1822    *
1823    *
1824    *           CATN IS A MACRO TO SEARCH FOR AN ENTRY WITH A GIVEN ENTRY
1825    *       NUMBER IN A CATALOG.  WHEN CALLING INDEX T POINTS TO A FILE CONTROL
1826    *       BLOCK FOR THE CATALOGED FILE WHOSE ENTRY IS BEING SOUGHT.  POSSIBLE
1827    *       STATUS RETURNS IN C$CSTAT ARE
1828    *            B$OK       ENTRY WAS FOUND
1829    *            B$RERR     IO ERROR IN READING CATALOG
1830    *            B$UERR     ENTRY WAS NOT FOUND OR CATALOG HAS BEEN DAMAGED
1831    *            B$FNC      FILE BECAME A SCRATCH FILE (NOT CATALOGED)
1832    *
1833    *
1834    CATN    MACRO   DA/NULL
1835            CRSM    SAVE,OFF
1836            IFE     '#1','',2       IF FIRST ARGUMENT IS NULL
1837            TSX0    C$CATNA         JUST CALL SUBROUTINE FOR SUCH
1838            IFE     1,2,6
1839            CATH    #1              READ IN HEADER OF CATALOG
1840            LDA     C$CSTAT         CHECK STATUS OF READ
1841            TNZ     *+4             BAD -- DON'T DO CATN
1842            LDA     C$HDA,T         GET DA AGAIN
1843            STC2    C$CATR,T        SET RETURN
1844            TRA     C$CATAC         JUMP INTO CATAC
1845            CRSM    RESTORE
1846            ENDM    CATN
```

C CATALOG OPERATIONS MACROS

```
1847                EJECT
1848    *
1849    *
1850    *           FUSE IS A MACRO WHICH FINDS OR CREATES AN ENTRY IN THE C$USE LIST
1851    *       FOR THE CURRENT ENTRY IN THE CURRENT CATALOG.  IT SETS THE C$USEFL FLAGS
1852    *       FOR THE ENTRY AND EXITS WITH A POINTER TO THE C$USE ENTRY IN C$USEC AND
1853    *       IN INDEX REGISTER X.
1854    *           ARGUMENT #1 IS THE ERROR RETURN IN CASE OF A C$USE LIST OVERFLOW.
1855    *       ARGUMENT #2, IF PRESENT, SPECIFIES A SEPARATE RETURN TO BE TAKEN IF A
1856    *       C$USE ENTRY ALREADY EXISTS AND WHICH DOES NOT UPDATE THE C$USEFL FLAGS.
1857    *       IF EITHER ARGUMENT IS "*ERROR" A ZOP FAULT IS GENERATED.                [09DEC79]
1858    *           FUSE PRESERVES INDEX REGISTER Y.
1859    *
1860    *
1861    *
1862    FUSE    MACRO   <OVERFLOW>,<DUPLICATE>                                         [09DEC79]
1863            IFE     '#1','*ERROR',2                                                [09DEC79]
1864            EAA     C$FUSED         ERROR ON C$USE LIST FULL                       [09DEC79]
1865            IFE     0,1,1                                                          [09DEC79]
1866            EAA     #1              RETURN ON C$USE LIST FULL                      [09DEC79]
1867            IFE     '#2','*ERROR',2                                                [09DEC79]
1868            ORA     C$FUSED,DL      ERROR ON DUPLICATE ENTRY                       [09DEC79]
1869            IFE     0,1,2                                                          [09DEC79]
1870            INE     '#2',''                                                        [09DEC79]
1871            ORA     #2,DL           RETURN ON DUPLICATE ENTRY                      [09DEC79]
1872            TSX0    C$FUSE          GO TO SUBROUTINE
1873            ENDM    FUSE
1874    *
1875    *
1876    *           DUSE IS A MACRO TO DELETE THE ENTRY REFERENCED BY INDEX REGISTER X
1877    *       FROM THE C$USE LIST.
1878    *
1879    *
1880    DUSE    MACRO
1881            TSX0    C$DUSE          USE SUBROUTINE
1882            ENDM    DUSE
1883    *
1884    *
1885    *           INSC IS A MACRO TO INSERT A NEW ENTRY INTO A CATALOG WHICH HAS
1886    *       BEEN READ INTO CORE BY THE CATS MACRO.  ON CALLING THE MACRO INDEX T
1887    *       POINTS TO THE CONTROLLING LIST ELEMENT FOR THE CATALOG OPERATION.
1888    *       THIS LIST ELEMENT IS LINKED IN TURN TO AN FCB FOR THE FILE TO BE ENTERED.
1889    *       UPON RETURN C$USE CONTAINS AN ENTRY FOR THE NEW CATALOG ENTRY AND
1890    *       F$CAT IN THE FILE FCB CONTAINS A POINTER TO THE C$USE ENTRY FOR IT.
1891    *       POSSIBLE STATUS RETURNS (IN THE A REGISTER) ARE
1892    *               B$OK            SUCCESSFUL
1893    *               B$QUEX          ALOC/MAX CHECK FAILED
1894    *               B$NSTOR         CORE CATALOG FULL, ALL HEADER DA'S USED, OR
1895    *                               TOO MANY ENTRIES IN THE CATALOG OR THE C$USE LIST
1896    *               B$UERR          BAD INFORMATION IN CATALOG
1897    *
1898    INSC    MACRO
```

C

```
                        CATALOG OPERATIONS MACROS

1899                    TSXO    C$INSC
1900                    ENDM    INSC
1901    *
1902    *
1903    *           RFCBC IS A MACRO WHICH UPDATES INFORMATION IN THE CURRENT CATALOG
1904    *       ENTRY.  ON CALLING INDEX T POINTS TO THE CONTROLLING LIST ELEMENT SET UP
1905    *       BY THE CATN MACRO.  THIS LIST ELEMENT IS LINKED IN TURN TO A FILE CONTROL
1906    *       BLOCK FOR THE FILE/CATALOG WHOSE ENTRY IS TO BE UPDATED.  RFCBC UPDATES
1907    *       CODED DATES AND ALSO C$N AND C$CDA IF THE APPEND PERMISSION BIT IS SET
1908    *       IN THE FILE CONTROL BLOCK.  UPON RETURN THE C$USE ENTRY FOR THE FILE WILL
1909    *       HAVE BEEN UPDATED IF THE CATALOG ENTRY HAS BEEN MOVED.  POSSIBLE STATUS
1910    *       RETURNS IN C$CSTAT ARE
1911    *            B$OK       SUCCESSFUL
1912    *            B$RERR     IO ERROR IN REPLACING ENTRY
1913    *            B$UERR     CATALOG HAS BEEN DAMAGED
1914    *            B$NSTOR    CORE CATALOG IS FULL OR DA LIST OVERFLOWED
1915    *
1916    *
1917    RFCBC   MACRO                       REPLACE FCB MACRO
1918            TSXO    C$RFCBC             REPLACE FCB SUBROUTINE
1919            ENDM    RFCBC
1920    ATACH   MACRO                       ATTACH TO FILE
1921            TSXO    C$ATTCH             USE THE SUBROUTINE
1922            ENDM    ATACH
1923    DTACH   MACRO                       MACRO TO CALL DETACH SUBROUTINE
1924            TSXO    C$DTACH             USE DTACH SUBROUTINE
1925            ENDM    DTACH
1926    *
1927    *
1928    *           THE GFCBC MACRO COPIES THE CURRENT ENTRY IN THE CATALOG INTO A FILE
1929    *       CONTROL BLOCK PROVIDING THAT THE NECESSARY ACCESS AND CONFLICT CHECKS ARE
1930    *       PASSED.  ON ENTRY INDEX T POINTS TO THE CONTROLLING LIST ELEMENT.  C$CACC
1931    *       IN THIS LIST ELEMENT CONTAINS THE ACCESSES DESIRED ON THE FILE (UPPER)
1932    *       AND THE ACCESSES REQUIRED ON THE FILE (LOWER).  IF ALL REQUIRED ACCESSES
1933    *       ARE AVAILABLE, A FILE CONTROL BLOCK FOR THE FILE WILL BE INSERTED IN
1934    *       THE LIST ELEMENT CHAIN BEHIND THE CONTROLLING LIST ELEMENT.  AN ENTRY IN
1935    *       THE C$USE LIST WILL HAVE BEEN MADE FOR THE FILE SHOWING THE ATTACHMENTS
1936    *       OBTAINED, AND C$USEC WILL POINT TO THIS ENTRY.  POSSIBLE STATUS RETURNS
1937    *       IN THE A REGISTER ARE AS FOLLOWS.
1938    *            B$OK       SUCCESSFUL -- A FILE CONTROL BLOCK WAS OBTAINED
1939    *            B$LOK      LOCKOUT -- A REQUIRED ACCESS WAS BUSY
1940    *            B$PRV      PROTECTION VIOLATION
1941    *            B$FAL      FAIL -- A REQUIRED ACCESS WAS NOT ALLOWED
1942    *            B$CLE      NO INFO IF CLIMB WITHOUT SEARCH
1943    *            B$QUEX     THE CATALOG ALOC EXCEEDED TWICE ITS MAX AND C$QUOTA WAS SET
1944    *            B$NSTOR    THE C$USE LIST IS FULL
1945    *
1946    GFCBC   MACRO                       GET FILE CONTROL BLOCK FOR CURRENT FILE
1947            TSXO    C$GFCB              TRANSFER TO SUBROUTINE
1948            ENDM    GFCBC
1949    *
1950    *
```

C CATALOG OPERATIONS MACROS

```
1951    *          RELC RELEASES A CATALOG, CAUSING IT TO BE WRITTEN OUT IF NECESSARY,
1952    *      AND STARTS THE NEXT CATALOG TASK.
1953    *
1954    *
1955    RELC   MACRO                    RELEASE CATALOG
1956           TSX0    C$RELC
1957           ENDM    RELC
1958    *
1959    *
1960    *          DELC DELETES THE CURRENT ENTRY FROM THE CURRENT CATALOG.
1961    *
1962    *
1963    DELC   MACRO
1964           TSX0    C$DELC           CALL SUBROUTINE
1965           ENDM    DELC
```

```
C                       MACROS

1966                    TTLS    MACROS                                                  [09DEC79]
1967    *
1968    *
1969    *
1970    *
1971    *               GTIM--LOADS TIME (TIMER UNITS) INTO A REG
1972    *
1973    GTIM    MACRO                   NO ARGUMENTS
1974            TSX0    X$GTIM          RETURN TIMER UNITS IN A
1975            ENDM    GTIM
1976    *
1977    * FOLLOWING MACRO IS USED TO COPY FROM/TO PROTECTED AREAS
1978    * 5 ARGS -- FCB-FROM,PTER-FROM,FCB-TO,PTER-TO,RETURN ADDRESS
1979    * USES SPECIAL FCBS AND POINTERS -- SEE BELOW
1980    * LENGTH OF COPY IN WORD FOLLOWING POINTER-FROM
1981    *
1982    OVCPY   MACRO   FCBF,PTRF,FCBT,PTRT,RET
1983            TSX0    OVCPY
1984            TRA     #5
1985            ZERO    #1,#2
1986            ZERO    #3,#4
1987            ENDM    OVCPY
1988    *
1989    *
1990    *
1991    *                               FREE
1992    *
1993    *       THIS MACRO AND SUBROUTINE RELEASES A CHANNEL WHICH
1994    *       WAS SIEZED VIA THE SIEZE MACRO. REGISTERS WILL BE DESTROYED
1995    *       ON EXIT, AND EXIT IS NOT GUARANTEED TO BE IMMEDIATE.
1996    *
1997    FREE    MACRO   'PUB'
1998            INE     '#1','PUB'      WE ONLY FREE PUBS
1999    ARGER.  SET     ARGER.#1        FLAG BAD ARGUMENTS
2000            TSX0    I$FREE                                                  [18AUG76]
2001            ENDM    FREE
2002    *
2003    *
2004    *                                                                       [18AUG76]
2005    *       MACRO TO SIEZE A SPECIFIC CHANNEL.  RETURNS TO                  [18AUG76]
2006    *            CALL+1     ILLEGAL CHANNEL                                 [18AUG76]
2007    *            CALL+2     XRP CONTAINS PUB INDEX                          [18AUG76]
2008    *            CALL+3     XRP CONTAINS PUB INDEX OF RELEASED CHANNEL      [17OCT76]
2009    *                                                                       [18AUG76]
2010    *       NOTE THAT A VALID LIST ELEMENT POINTER MUST BE STORED IN        [18AUG76]
2011    *       Q$BUSY+P$Q,P AND THE FREE MACRO INVOKED UNDER THE CONTROL       [18AUG76]
2012    *       OF THAT LIST ELEMENT IN ORDER TO RELEASE A CHANNEL SIEZED BY    [18AUG76]
2013    *       THIS MACRO.                                                     [18AUG76]
2014    *                                                                       [18AUG76]
2015    *       MAY DESTROY ANY REGISTER BUT XRT.                               [18AUG76]
2016    *                                                                       [18AUG76]
2017    CHAN    MACRO   (CHANNEL#)                                              [18AUG76]
```

C

```
                         MACROS
2018                     EAA     #1                                          [18AUG76]
2019                     TSXO    I$CHAN                                      [18AUG76]
2020                     ENDM    CHAN                                        [18AUG76]
2021     *                                                                   [18AUG76]
2022     *                                                                   [170CT76]
2023     *       RFCB & RFCBA  --  THESE TWO MACROS AND THE CORRESPONDING    [170CT76]
2024     *       SUBROUTINES REPLACE THE FCB OF THE FILE REFERENCE NUMBER    [170CT76]
2025     *       IN REGISTER 'P' WITH THE FCB IN THE LIST ELEMENT POINTED    [170CT76]
2026     *       TO BY 'T'.  THE LIST ELEMENT IS THEN DEALLOCATED.  RFCB     [170CT76]
2027     *       REPLACES THE ENTIRE FCB;  RFCBA REPLACES ALL BUT F$ABORT     [170CT76]
2028     *       AND IS CALLED AT THE CONCLUSION OF A COPY.                  [170CT76]
2029     *                                                                   [170CT76]
2030     RFCB    MACRO                                                       [170CT76]
2031             TSXO    $RFCB           CALL THE SUBROUTINE                 [170CT76]
2032             ENDM    RFCB                                                [170CT76]
2033     *
2034     *
2035     *       FCBLIS COPIES A FILE CONTROL BLOCK POINTED TO BY P TO A LIST ELEMENT.
2036     *       IF P IS ZERO, THEN A CORE FCB IS COPIED IF #1 IS NULL AND A FCB FOR
2037     *       THE MFD IS COPIED IF #1 = MFD.
2038     *
2039     FCBLIS  MACRO
2040             IFE     '#1','MFD',2
2041             TSXO    E$FCBLM         SUBROUTINE ENTRY FOR MFD COPY
2042             IFE     0,1             OTHERWISE
2043             TSXO    E$FCBLS         SUBROUTINE ENTRY FOR CORE FCB
2044             ENDM    FCBLIS
2045     *
2046     *
2047     DEALOC  MACRO
2048             LDA     K$DLC,DL        GET ADD OF ROUTINE
2049             MTQA                    QUEUE THE TASK
2050             LDX     T,T$LINK,T      POP T
2051             ENDM    DEALOC
2052     *
2053     *
2054     SPEBLK  MACRO                   CONDITIONALLY UNROADBLOCKS JOB
2055             TSXO    E$EQNCQ         WAKE JOB UP IF NECESSARY
2056             ENDM    SPEBLK
2057     *
2058     *
2059     *       THE FOLLOWING MACROS  ARE USED TO ADD OR DELETE JOBS FROM THE
2060     *       CORE QUEUE OR NON-CORE QUEUE.  THE EACH TAKE AS THE FIRST ARGUMENT
2061     *       EITHER 'CQ' OR 'NCQ'.  IN ADDITION, IF THE SECOND ARGUMENT TO EQJ
2062     *       IS NOT NULL, THE JOB WILL BE ADDED TO THE FRONT OF THE SPECIFIED
2063     *       QUEUE.
2064     *
2065     EQJ     MACRO   QUEUENAME
2066             IFE     '#2','',6
2067             LDX     X,$#1I          SET X TO INSERT JOB NUMBER
2068             STX     J,J$CQ,X        STORE J IN THE END OF THE CHAIN
2069             LDX     X,0,DU          SET LAST LINK
```

C MACROS

```
2070                STX     X,J$CQ,J        EQUAL TO ZERO
2071                STX     J,$#1I          UPDATE INSERT POINTER TO J
2072                INE     '#2','',5
2073                LDX     X,$#1F          GET POINTER TO FIRST JOB
2074                TNZ     *+2             SKIP IF QUEUE NOT EMPTY
2075                STX     J,$#1I          UPDATE INSERT POINTER IF QUEUE EMPTY
2076                STX     X,J$CQ,J        LINK FORMER JOB TO J
2077                STX     J,$#1F          MAKE J FIRST
2078                ENDM    EQJ
2079    *
2080    *
2081    DQJ         MACRO   QUEUENAME
2082                CRSM    SAVE,ON
2083                EAX     Y,$#1BEG        GET RELATIVE BEGINNING OF QUEUE
2084    #4          CMPX    J,J$CQ,Y        SEE IF IT POINTS TO THIS JOB
2085                TZE     #5              YES, GO UNLINK JOB
2086                LDX     Y,J$CQ,Y        STEP ALONG THE CHAIN
2087                TNZ     #4              CONTINUE IF MORE JOBS
2088                TRA     #6              EXIT IF J NOT ON THE QUEUE
2089    #5          LDX     X,J$CQ,J        GET LOOKAHEAD POINTER
2090                STX     X,J$CQ,Y        UNLINK JOB
2091                TNZ     #6              EXIT IF NOT LAST
2092                STX     Y,$#1I          ELSE UPDATE THE INSERT POINTER
2093    #6          NULL
2094                CRSM    RESTORE
2095                ENDM    DQJ
2096    *
2097    *
2098    ENQJ        MACRO   LENGTH,JUMP  LOC     OR NULL
2099                CRSM    SAVE,OFF
2100                INE     '#1','',4                                      [17OCT76]
2101                IFE     #1,1,2                                         [17OCT76]
2102                GETD    1,NBUG                                         [17OCT76]
2103                IFE     0,1,1                                          [17OCT76]
2104                GETD    #1              GET LIST ELEMENT FOR MLINK TASK
2105                IFE     '#2',''
2106                ERROR   !
2107                LDQ     #2,DL           GET RESTART ADDRESS
2108                TSX0    E$ENQS          QUEUE FOR JOB
2109                CRSM    RESTORE
2110                ENDM    ENQJ
```

C MACROS

```
2111            EJECT                                                   [09DEC79]
2112    *
2113    *
2114    *           TREE IS A MACRO WHICH CALLS A SUBROUTINE OF THE SAME NAME
2115    *       TO SEARCH THE JOB TREE FOR JOBS RUNNING BELOW THE JOB WITH NUMBER J.
2116    *       EACH TIME SUCH A JOB IS FOUND A TSX0 #1 IS EXECUTED, SO THAT A
2117    *       SUBROUTINE CAN BE INSERTED TO PERFORM ANY DESIRED FUNCTION.
2118    *       RETURNING BY A TRA 0,0 WILL CONTINUE THE SEARCH THROUGH THE JOB
2119    *       TREE, WHILE A TRA 1,0 WILL CAUSE THE SUBTREE BELOW THE JOB JUST
2120    *       FOUND TO BE IGNORED.
2121    *
2122    *
2123    TREE    MACRO   ADDRESS  OF SUBROUTINE
2124            EAX     X,#1            SUBROUTINE ADDRESS TO X
2125            TSX0    E$TREE          GO TO TREE ROUTINE
2126            ENDM    TREE
2127    *
2128    DECCT   MACRO   COUNTER         DECREASE COUNT
2129            LCA     1,DL            GET A MINUS ONE
2130            ASA     #1              ADD TO APPROPRIATE COUNTER
2131            TMI     $ZOPF,*         ERROR IF IT WENT NEGATIV
2132            ENDM    DECCT
```

C PAGE TABLE SIZE DEFINITIONS

```
                    2133            TTLS    PAGE TABLE SIZE DEFINITIONS
                    2134    *
                    2135    *
                    2136    *
                    2137            HEAD    K,X
            000004  2138    TLEN    EQU     4
            000300  2139    PLEN    EQU     192                                    [08AUG77]
                    2140    *
                    2141    *
                    2142    *       DEFINITION FOR  STORAGE ALLOCATION LIST ELEMENTS
                    2143    *
                    2144            HEAD    K
                    2145
            000001  2146    TABLE   EQU     1
            000002  2147    PLACE   EQU     2
            000003  2148    NDA     EQU     3
            000004  2149    TYPE    EQU     4
            000005  2150    RET     EQU     5
            000005  2151    SIZE    EQU     5
            000006  2152    LOWER   EQU     6
            000007  2153    UPPER   EQU     7
```

K PIO MACRO

```
2154            TTLS    PIO MACRO                                                   [21APR77]
2155    *                                                                           [21APR77]
2156    *       THE FOLLOWING MACRO CALLS THE PHYSICAL I/O ROUTINE X$PIO            [21APR77]
2157    *                                                                           [21APR77]
2158    PIO     MACRO   PDA*,DCW*,COMMAND*,STATUS*,CPC/MPC,TIMOUT  TIME             [21APR77]
2159            REM                                                                 [21APR77]
2160            TSX0    X$PIO                                                       [21APR77]
2161            ARG     #1              PTR TO PDA                                  [21APR77]
2162            ARG     #2              PTR TO DCW                                  [21APR77]
2163            ARG     #3              PTR TO COMMAND                              [21APR77]
2164            ARG     #4              PTR TO STATUS PAIR                          [21APR77]
2165            IFE     '#5',',2                                                    [21APR77]
2166            ARG     1,DU            ASSUME MPC CHANNEL                          [21APR77]
2167            IFE     0,1,7                                                       [21APR77]
2168            IFE     '#5','CPC',2                                                [21APR77]
2169            ARG     0,DU            SET CPC CHANNEL                             [21APR77]
2170            IFE     0,1,4                                                       [21APR77]
2171            IFE     '#5','MPC',2                                                [21APR77]
2172            ARG     1,DU            SET MPC CHANNEL                             [21APR77]
2173            IFE     0,1,1                                                       [21APR77]
2174            ARG     #5              POINT TO CHANNEL TYPE                       [21APR77]
2175            IFE     '#6',''                                                     [21APR77]
2176            ARG     10,DL           SET DEFAULT 10 SECOND TIMEOUT TIME          [21APR77]
2177            INE     '#6',''                                                     [21APR77]
2178            ARG     #6              SET TIMEOUT TIME                            [21APR77]
2179            ENDM    PIO                                                         [21APR77]
```

K XLOG MACRO

```
2180                TTLS    XLOG MACRO                                                  [21APR77]
2181    *                                                                               [21APR77]
2182    *                                                                               [21APR77]
2183    *           LOG MACRO -- SEE LOG ROUTINE FOR CALLING SEQUENCE                   [21APR77]
2184    *                                                                               [21APR77]
2185    XLOG        MACRO   (TEXT),ARG1,ARG2,...,ARG8                                  [21APR77]
2186                TSX0    X$LOG                                                       [21APR77]
2187                CRSM    SAVE,ON                                                     [21APR77]
2188                PMC     SAVE,OFF            DISABLE MACRO LISTING
2189                ZERO    #61,#62                                                     [21APR77]
2190    LOGSET      SET     *                                                           [21APR77]
2191                INE     '#1',''                                                     [21APR77]
2192                UTEXT   \#1\                                                        [21APR77]
2193    #62         EQU     *-LOGSET                                                    [21APR77]
2194                CRSM    SAVE,OFF                                                    [21APR77]
2195                PMC     RESTORE             ENABLE MACRO LISTING
2196                INE     '#2',''                                                     [21APR77]
2197                ARG     #2                                                          [21APR77]
2198                INE     '#3',''                                                     [21APR77]
2199                ARG     #3                                                          [21APR77]
2200                INE     '#4',''                                                     [21APR77]
2201                ARG     #4                                                          [21APR77]
2202                INE     '#5',''                                                     [21APR77]
2203                ARG     #5                                                          [21APR77]
2204                INE     '#6',''                                                     [21APR77]
2205                ARG     #6                                                          [21APR77]
2206                INE     '#7',''                                                     [21APR77]
2207                ARG     #7                                                          [21APR77]
2208                INE     '#8',''                                                     [21APR77]
2209                ARG     #8                                                          [21APR77]
2210                INE     '#9',''                                                     [21APR77]
2211                ARG     #9                                                          [21APR77]
2212                CRSM    RESTORE                                                     [21APR77]
2213                PMC     SAVE,OFF            DISABLE MACRO LISTING
2214    #61         NULL                                                                [21APR77]
2215                PMC     RESTORE             ENABLE MACRO LISTING
2216                CRSM    RESTORE                                                     [21APR77]
2217                ENDM    XLOG                                                        [21APR77]
```

K PIO INITIALIZATION COMM AREA DEFINITIONS

```
                    2218            TTLS    PIO INITIALIZATION COMM AREA DEFINITIONS                    [21APR77]
                    2219    *                                                                           [21APR77]
                    2220    *                                                                           [21APR77]
                    2221    *                                                                           [21APR77]
                    2222    *       THE FOLLOWING SYMBOLS DEFINE OFFSETS IN THE PIO INIT COMM AREA.     [21APR77]
                    2223    *       THIS COMM AREA SHOULD BE SET UP BEFORE CALLING PIONIT.              [21APR77]
                    2224    *                                                                           [21APR77]
000000              2225    COMMBX  EQU     0               MBX BASE/PORT NUMBER FOR IOM # 0            [21APR77]
000001              2226            EQU     1               MBX BASE/PORT NUMBER FOR IOM # 1            [21APR77]
000002              2227            EQU     2               MBX BASE/PORT NUMBER FOR IOM # 2            [21APR77]
000003              2228            EQU     3               MBX BASE/PORT NUMBER FOR IOM # 3            [21APR77]
000004              2229    COMIMW  EQU     4               (UPPER) ADDRESS OF IMW AREA                 [21APR77]
000004              2230    COMIOM  EQU     4               (LOWER) IOM NUMBER OF BOOT IOM              [21APR77]
000005              2231    PDACON  EQU     5               (UPPER) PDA FOR CONSOLE                     [21APR77]
000006              2232    PDARDR  EQU     6               (UPPER) PDA FOR READER                      [21APR77]
000007              2233    PDATAP  EQU     7               (UPPER) PDA FOR TAPE                        [21APR77]
000010              2234    PDAENV  EQU     8               (UPPER) PDA FOR ENV DECK DEVICE             [21APR77]
000011              2235    COMARL  EQU     9               LENGTH OF COMM AREA                         [21APR77]
                    2236    *                                                                           [21APR77]
                    2237            UPPER   COMIMW,PDACON,PDARDR,PDATAP,PDAENV                          [09DEC79]
                    2238            LOWER   COMIOM                                                     [09DEC79]
                    2239    *                                                                           [09DEC79]
                    2240    *
                    2241    *
                    2242    *       END OF INSERT FILE
                    2243    *
                    2244            HEAD                                                               [09DEC79]
                    2245    *
```

```
                                        ** PRODUCT TRACKING AND GENERAL INFO DEFINITIONS

                    2246                TTLS    ** PRODUCT TRACKING AND GENERAL INFO DEFINITIONS
                    2247    *
                    2248    *           ** GENERAL INFORMATION BLOCK
                    2249    *
                    2250                HEAD    W
                    2251    *
000000              2252    GIFLG       EQU     0               ** GENERAL INFORMATION FLAGS
000001              2253    GIPUN       EQU     GIFLG+1         ** PRODUCT TRACKING STARTING UNITS
000002              2254    GIPBT       EQU     GIPUN+1         ** PRODUCT TRACKING STARTING BTIME
000003              2255    GIPTK       EQU     GIPBT+1         ** PRODUCT TRACKING CONTROL AND USER INFO
                    2256    *
000014              2257    GIBKL       EQU     GIPTK+9         ** LENGTH OF GENERAL INFORMATION BLOCK
                    2258    *
                    2259    *
                    2260    *           ** GENERAL INFORMATION FLAG DEFINITIONS
                    2261    *
400000              2262    GIFPT       BOOL    400000          ** PRODUCT TRACKING IN PROGRESS
                    2263    *
                    2264    *
                    2265    *           ** PRODUCT TRACKING BLOCK DEFINITION
                    2266    *
000001              2267    PTBAS       EQU     Q$RUN+1         ** STARTING LOCATION
000002              2268    PTUSR       EQU     PTBAS+1         ** USER NUMBER (2 WORDS)
000004              2269    PTGIF       EQU     PTUSR+2         ** GENERAL INFORMATION FROM GIF BLOCK
000020              2270    PTEUN       EQU     PTGIF+GIBKL     ** ENDING UNITS
000021              2271    PTEBT       EQU     PTEUN+1         ** FINAL B-TIMER
000022              2272    PTEWC       EQU     PTEBT+1         ** FINAL WALL CLOCK
000023              2273    PTEDT       EQU     PTEWC+1         ** DATE (2 WORDS)
000025              2274    PTFAC       EQU     PTEDT+2         ** STATE VECTOR CPFAC,IOFAC,SFAC,PID(3),UMPY
000034              2275    PTCLN       EQU     PTFAC+7         ** CORE LENGTH AT END
                    2276    *
000035              2277    PTEND       EQU     PTCLN+1         ** LENGTH OF ENTIRE BLOCK
000034              2278    PTBKL       EQU     PTEND-PTBAS     ** ACTUAL PRODUCT TRACK INFO
                    2279    *
                    2280    *
                    2281    *
000000              2282            THE END
```

CROSS REFERENCE TABLE

4000	B	AP	630	633	
10000	B	EX	629	633	
1000	B	RD	632	633	
2000	B	WT	631	633	
400000	B	CAP	684	688	
200000	B	CFC	645	663	
100000	B	CFD	646	663	
40000	B	CFR	647	663	
200000	B	CWT	685	688	689
2	B	DFE	764	765	
10000	B	MDA	554	570	
40000	B	OWN	627	633	
1	B	SFE	763	765	
10000	B	CFCL	650	663	
20000	B	CFGA	649	663	
400	B	NTPD	673	675	
400000	B	NTPS	671	675	
200	B	RSVD	674	675	
200000	B	RSVS	672	675	
400000	B	SWAP	549	570	
400	BC	FRVM	654	663	
4000	BS	WREQ	555	570	
10	C	CATR	1620	1639	
4	C	CLEN	467	485	
6	C	DALT	479	485	
11	C	FLAG	1623	1640	
11	C	PERM	1622	1639	
22	C	TACC	1636	1640	
6	C	TYPE	480	486	
10	C	USEP	1621	1640	
22	C	CBITS	1634	1635	1639
5	C	ENTRY	469	486	
5	C	INDEX	468	485	
22	C	MMENO	1635	1639	
21	C	NAMND	1632	1633	1639
21	C	NAMPT	1633	1640	
4	COMI	MW	2229	2237	
4	COMI	OM	2230	2238	
17	C	PASLE	1630	1640	
17	C	PASPT	1629	1639	
1	F	J	528	531	
1	F	FR	529	532	
0	F	ACC	371	418	
0	F	BIT	521	531	
6	F	DFR	414	419	
1	F	RET	376	419	
6	F	SFR	413	414	418
0	F	LINK	523	532	
0	F	TYPE	372	419	
1	FA	BORT	373	418	
2	I	CMD	867	869	
10	I	DAC	879	880	

CROSS REFERENCE TABLE

1	I DEV	865	887			
3	I PUB	869	870	871	872	887
15	I IDCW	884	885			
7	I MODE	877	878	879	887	
3	I PUBL	870	888			
1	I TYPE	866	867	888		
6	I URET	874	876	877	887	
6	IADEXT	876	888			
12	IDCWWD	881	882			
7	IMODEL	878	888			
11	IQUEWD	880	881			
5	IQWORD	873	874			
4	ISEKAD	872	873			
13	ISIDCW	882	833			
14	ISKDCW	883	884			
3	ISPRET	871	888			
20	M EIS	272	278			
4000	M NSA	265	278			
400	MCACHE	268	278			
200	MEXMEM	269	278			
7740	MFTVMK	262	277			
4	MOVRLP	274	278			
3	MPROCN	275	278			
2000	MSER66	266	278			
10	MSLMEM	273	278			
40000	MTSOPT	264	278			
2	NPROS	87	138			
5	PDACON	2231	2237			
10	PDAENV	2234	2237			
6	PDARDR	2232	2237			
7	PDATAP	2233	2237			
0	Q RUN	835	836	839	2267	
777777	Q BUSY	1024	1029			
0	Q LINK	834	838			
777777	Q NPRI	1023	1028			
30	S IC	320	321			
47	S BIT	340	341	367		
73	S FCB	361	363			
64	S PID	356	357			
0	S REG	317	318			
45	S TIO	337	338			
10	S AREG	318	319			
31	S BARS	321	322			
35	S BUSY	328	329	367		
41	S CATW	333	334			
55	S CLEN	346	347			
60	S FREE	350	353			
36	S FRUN	330	331	367		
57	S HOLE	348	350	366		
47	S INTP	339	340	366		
50	S JMEM	341	342			
42	S SCRW	334	335			

CROSS REFERENCE TABLE

63	S SFAC	355	356	
56	S SPEC	347	348	
36	S SWAP	329	330	366
44	S TCPU	336	337	
67	S UMPY	357	358	
54	SCORET	345	346	
61	SCPFAC	353	354	
35	SFTYPE	327	328	366
53	SIOCHG	344	345	
62	SIOFAC	354	355	
71	SIOTIM	359	360	
70	SIOUCH	358	359	
40	SJACES	332	333	
43	SJTIME	335	336	
32	SLIMIT	323	324	367
72	SPTIMR	360	361	
20	SPTLEN	319	320	
37	SQUANT	331	332	
52	SSTIME	343	344	
51	SSVMEM	342	343	
32	STACES	322	323	366
46	STCORE	338	339	
33	STIMER	324	325	
34	SUTYPE	325	327	366
777777	T LEN	845	848	
777777	T LINK	844	847	
1340	V IMW	125	918	
1400	V MBX	127	915	
2200	V FTVS	136	138	
2300	VCFILE	138	140	1002
2340	VSISTK	140	921	
14	WGIBKL	2257	2270	
0	WGIFLG	2252	2253	
2	WGIPBT	2254	2255	
3	WGIPTK	2255	2257	
1	WGIPUN	2253	2254	
1	WPTBAS	2267	2268	2278
34	WPTCLN	2275	2277	
21	WPTEBT	2271	2272	
23	WPTEDT	2273	2274	
35	WPTEND	2277	2278	
20	WPTEUN	2270	2271	
22	WPTEWC	2272	2273	
25	WPTFAC	2274	2275	
4	WPTGIF	2269	2270	
2	WPTUSR	2268	2269	
1400	X MBX	915	924	
1412	X PCWA	924	925	
10	XCONCH	897	924	

MACRO CROSS REFERENCE TABLE

0	ALARM	1538
0	ALC	1105
0	APROC	1140
0	ATACH	1920
0	BUG	1216
0	BUGA	1226
0	CATC	1759
0	CATH	1782
0	CATL	1813
0	CATN	1834
0	CHAN	2017
0	CKPT	1260
0	COPY	1568
0	DABL	1178
0	DEALOC	2047
0	DECCT	2128
0	DELC	1963
0	DEQ	1361
0	DQJ	2081
0	DTACH	1923
0	DUSE	1880
0	ENABL	1184
0	ENQ	1310
0	ENQJ	2098
0	EQJ	2065
0	EXPAND	1487
0	FCBLIS	2039
0	FCBPNT	1672
0	FREE	1997
0	FUSE	1862
0	GET	1414
0	GETB	1436
0	GETBQ	1458
0	GETD	1424
0	GFCBC	1946
0	GFDA	1768
0	GFR	1680
0	GTIM	1973
0	IFIOC	75
0	IFIOM	71
0	INSC	1898
0	INVERT	1129
0	LOG	1503
0	LOGF	1521
0	MTASK	1373
0	MTQ	1338
0	MTQA	1347
0	ORDER	1556
0	OVCPY	1982
0	PIO	2158
0	PROS	1162
0	PROTO	1122

MACRO CROSS REFERENCE TABLE

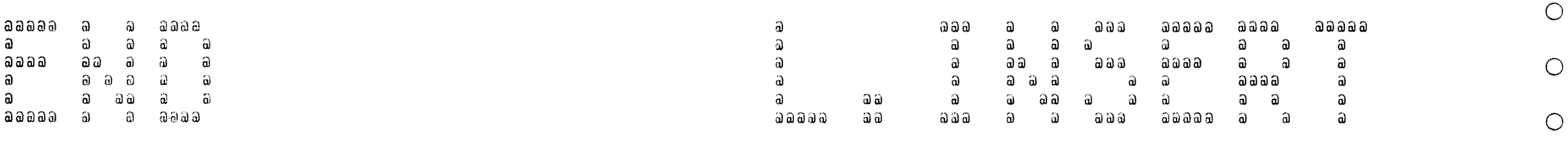
0	PTQ	1296
0	QLOCK	1687
0	QNLOCK	1691
0	QUEUE	1275
0	REL	1467
0	RELC	1955
0	RFCB	2030
0	RFCBC	1917
0	SHRINK	1474
0	SPEBLK	2054
0	SPROC	1153
0	TCATL	1818
0	TREE	2123
0	WSC	1708
0	XLOG	2135

OPCODE CROSS REFERENCE TABLE

1	MLDA	1799	
2	RMCM	1179	1185
2	SMCM	1181	1187

THERE WEREN'T ANY WARNING FLAGS IN THIS ASSEMBLY
0 IS THE NEXT AVAILABLE LOCATION
19 K CORE USED IN THIS ASSEMBLY

B**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB* COMPUTER SHARING SERVICES *BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**B
B**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB* COMPUTER SHARING SERVICES *BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**B



B**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB* COMPUTER SHARING SERVICES *BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**BB**B
241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241 241

09/24/81 14:17:13 PRINTOUT #241