LABEL 000000000PRINTER001751J0CC EX OBJECT/READ;FILE SOURCEFILE=SYMBOL/MLOGAN←←0000000 OBJECT /READ

SYMBOL/MLOGAN

```
COMMENT: * TITLE: B5500/B5700 MARK XIV SYSTEM RELEASE                    *     00000010
         * FILE ID: SYMBOL/MLOGAN  TAPE ID: SYMBOL2/FILE000              *     00000011
         * THIS MATERIAL IS PROPRIETARY TO BURROUGHS CORPORATION         *     00000012
         * AND IS NOT TO BE REPRODUCED, USED, OR DISCLOSED               *     00000013
         * EXCEPT IN ACCORDANCE WITH PROGRAM LICENSE OR UPON             *     00000014
         * WRITTEN AUTHORIZATION OF THE PATENT DIVISION OF               *     00000015
         * BURROUGHS CORPORATION, DETROIT, MICHIGAN 48232                *     00000016
         *                                                               *     00000017
         * COPYRIGHT (C) 1972 BURROUGHS CORPORATION                      *     00000018
         *  AA320206   AA386657                                          *;    00000019
                                                                               00000020
% COMMENT DATE 21 AUG 72  :  LATEST AND GREATEST;                              00000100
% COMMENT DATE 14 DEC 73  :  REVISED TO HANDLE EXCEPTION CONDITIONS;           00000110
%                                                                              00000120
% VARIOUS REPORTS ARE PRODUCED BY THIS PROGRAM KEYED ON THE COMMON VALUE       00000200
% THE COMMON VALUE HAS AN INPUT PART AND AN OUTPUT PART...                     00000210
% THE UNITS DIGIT SPECIFIES THE INPUT MODE AS FOLLOWS:                         00000300
%     1 = SPO INPUT                                                            00000310
%     2 = CARD INPUT                                                           00000320
% THE TENS DIGIT SPECIFIES WHICH REPORTS TO GENERATE AS FOLLOWS:               00000400
%     0 = GENERATE ANY AND ALL REPORTS.                                        00000410
%     1 = CHRONOLOGICAL LISTING REPORT ONLY.                                   00000420
%     2 = SUMMARY REPORTS FOR PERIPHERIAL UNIT ERRORS ONLY.                    00000430
%     4 = REAL TIME TAPE WRITE FAILURE REPORT ONLY.                            00000440
%     8 = ONLINE CONFIDENCE TEST RESULT SUMMARIES ONLY.                        00000450
% TO GENERATE COMBINATIONS OF REPORTS, SIMPLY ADD UP THE TENS DIGIT           00000500
% FOR EACH REPORT DESIRED, FOR EXAMPLE:                                       00000510
%     3 = SUMMARY REPORTS AND CHRONOLOGICAL LISTING REPORT...                  00000520
%     5 = CHRONOLOGICAL LISTING REPORT AND REAL TIME TAPE TEST REPORT...       00000530
%     6 = SUMMARY REPORTS AND REAL TIME TAPE TEST REPORT....                   00000540
%    10 = SUMMARY REPORTS OF PERIPHERIAL UNIT AND ONLINE MAINT ENTRIES..       00000550
%                                                                             00000600
% TO MAKE THE PROGRAM EASIER TO RUN FOR NORMAL TERMINATION PROCEDURES         00000610
% THE FOLLOWING MODIFICATIONS HAVE BEEN MADE TO THE COMMON VALUES:            00000612
%                                                                             00000614
% IF THE PROGRAM IS EXECUTED WITH A USERCODE OF "SITE" AND                    00000616
% THE COMMON VALUE IS OF THE FORM:                                            00000618
%     COMMON = 0    OR    COMMON = MMDD     (MM=MONTH, DD=DAY)               00000620
% THEN THE FOLLOWING OPERATIONS OCCUR:                                        00000622
% (IF THE USERCODE IS NOT "SITE" AND COMMON =0 OR COMMON= MMDD                00000624
% THEN THE NORMAL OPERATION OF THE COMMON VALUES IS USED)                     00000626
%                                                                             00000628
% 1) ONLY THE SUMMARY REPORTS FOR PERIPHERIAL UNIT ERRORS                     00000632
% ARE PRODUCED,                                                               00000634
%                                                                             00000635
% 2) WITH A COMMON VALUE OF 0, ALL MAINTENANCE LOGS CREATED                   00000636
% ON THE DAY OF EXECUTION TOGETHER WITH THE CURRENT MAINT/LOG                 00000638
% ARE USED,                                                                   00000640
%                                                                             00000642
% 3) IF THE COMMON VALUE MMDD IS USED, ONLY THE MAINTENANCE                   00000644
% LOG FILES CREATED ON THE DATE "MMDD" WHICH ARE ON THE DISK                  00000646
% ARE USED.                                                                   00000648
%                                                                             00000650
% SAMPLE EXECUTE                                                             00000652
% Q USER = SITE ; EXECUTE LOGANL/MAINT; COMMON = 0 ; END.                     00000654
% Q USER = SITE ; EXECUTE LOGANL/MAINT; COMMON = 0622 ; END.                  00000656
%                                                                             00000658
BEGIN                                                                        00001000
INTEGER COMMON;                                                              00002000
DEFINE                                                                       00002100
```

```
      LOGFID="MNTLOG "#, % CHANGE THIS FOR SHAREDISK                    00002200
      MAXFNUM=99#,       % MAXIMUM NUMBER OF FILES                     00002300
      MAXENTRY=205#;     % MAXIMUM SIZE OF LOG ENTRY                   00002400
1     REAL                                                             00003000
2          I,J,K,N,                                                    00004000
3          MFID,FID,ALPHA1,ALPHA2,          %KEEP IN THIS ORDER FOR SCAN 00005000
4          TAPERRORS,DISKERRORS,TAPETESTS,                             00006000
5          LUN,TYPE,FNUM,MIX,MARKER,                                   00007000
6          LASTENTRY,LOGENTRY,NUMENTRIES,RCNT,SEGS,                    00008000
7          STARTDATE,STARTIME,STOPDATE,STOPTIME,                       00009000
8          CDATE,ACTDATE,TIME0,TIME1,PROCTIME,IOTIME,                  00010000
9          ETYPE,EUSPEED,T1,T2,LUNMAX,MIXMAX;                          00011000
10    ARRAY                                                            00012000
11         A[0:9],                                                     00012100
12         B[0:209],                                                   00012200
13         DAYS[0:6],                                                  00012300
14         INX[0:39],                                                  00012400
15         JAR[0:31,0:299],                                            00012500
16         JUNKBUFFER[0:39],                                           00012600
17         L[0:17],                                                    00012700
18         MESS[0:25],                                                 00012800
19         RETRY[0:31],                                                00012900
20         TABLE[0:39,0:1022],                                         00013000
21         TERMINALMESSAGA[0:9],                                       00013100
22         TINU[0:31],                                                 00013200
23         TPECNF[0:1,0:31],                                           00013300
24         UNIT[0:63],                                                 00013400
25         UNK[0:11],                                                  00013500
26         UTITLES[0:17];                                              00013600
27    BOOLEAN                                                          00021000
28         LISTOG, SUMTOG, TESTOG, ONLTOG, % REPORT TOGGLES            00021100
29         DRUMTOG, SUMFILETOG, CROSSTOG, HALTOG, XDTOG,               00021200
30         INPASS1,MAINTLOG,USEFINPUT;                                 00021300
31    LABEL                                                            00022000
32         STARTUP,FINISHUP,RENTER,ENDOFILE,EOP,FINIS,                 00022100
33         FLAGBIT,INVALDINDEX,INTEGROVFLW,EXPONOVFLW,DIVBYZERO;       00022200
34    MONITOR FLAG,INDEX,INTOVR,EXPOVR,ZERO:=DIVZERO;                  00022300
35    FILE                                                             00023000
36         LOG     DISK SERIAL                          (2,5,90),      00024000
37         FINPUT  DISK SERIAL[20:30]                   (2,10,30),     00024100
38         PBD     DISK SERIAL[20:1500] "PBD""MAINT"    (2,18,90),     00025000
39         TST     DISK SERIAL[20:100]  "TST""MAINT"    (1,210,210),   00025100
40         TEMP    DISK SERIAL[20:600]  "TMP""MAINT"    (2,10,30),     00026000
41         BADISK  DISK SERIAL[20:600]  "XD" "MAINT"    (2,10,30),     00026100
42         SUM     DISK SERIAL[20:600]  "SUM""MAINT"    (2,10,30),     00027000
43         SORTED  DISK SERIAL[20:600]  "SRT""MAINT"    (2,10,30),     00028000
44         SPO      11(1,10),                                          00029000
45         LPA      17(3,15),                                          00029100
46         WHATLOG   0(2,10);                                          00030000
47    SWITCH FILE SWIN  := TEMP,BADISK;                                00030100
48    SWITCH FILE SWOUT := SUM,SORTED;                                 00031000
49    DEFINE                                                           00032000
50         COM   =T1#,                                                 00032100
51         CF=33:15#, FF=18:15#, SF=8:10#,                             00032200
52         ARROW ="←"#,                                                00033000
53         BIT   =              IF SB THEN DS:=SET ELSE DS:=RESET;SKIP SB#, 00034000
54         OCTADE=DS:=3 RESET;3(IF SB THEN DS:=SET ELSE DS:=RESET;SKIP SB)#, 00035000
55         BL    =BOOLEAN#,                                            00036000
56         PRINTMFID  = WRITE(LPA[DBL],FMFID,LMFID) #,                 00036010
57         WRITESINGLE= L[17]:=0; WRITE(PBD,18,L[*])#,                 00037000
```

```
        WRITEDOUBLE= L[17]:=1; WRITE(PBD,18,L[*])#,                          00038000
        WRITEPAGE  = L[17]:=999; WRITE(PBD,18,L[*])#,                        00039000
        WRITESPACES(WRITESPACES1)=                                           00040000
                   L[17]:=WRITESPACES1; WRITE(PBD,18,L[*])#,                 00041000
        SINGLESPACE= L[17]=0#,                                               00042000
        DOUBLESPACE= L[17]=1#,                                               00043000
        PAGEIT     = L[17]=999#,                                             00044000
        EOFPBD     = L[17]=1023#,                                            00045000
        ZERO(ZERO1,ZERO2)=ZEROUT((ZERO1)-1,ZERO2)#,                          00045100
        DUMMY=DUMMY#;                                                        00046000
LIST                                                                         00046100
     LMFID(MFID,[6:6],MFID,FID,[6:6],FID),                                   00046200
     LALPHA(ALPHA1,ALPHA2,NUMENTRIES,I:=LOGENTRY,[CF],I DIV 6);               00046300
FORMAT                                                                       00047000
     SPACES(*(/)),                                                           00048000
     FMFID("ANALYSIS FOR FILE  ",A1,A6,"/",A1,A6),                           00049000
     FDATE(X8,"DATE IS   ",A5,X2,A6,"DAY,   ",0),                            00050000
     FTERMINATE(A6,A6," BRANCH : ENTRY ",O,"AT RECORD ",O,"SEGMENT ",O);     00051000
%                                                                            00065000
STREAM PROCEDURE BOJMESS                                                     00066000
     (L,JULIAN,DAY,DATE,TIME,LISTOG,SUMTOG,TESTOG,ONLTOG);                   00066100
  VALUE JULIAN,DAY,DATE,TIME,LISTOG,SUMTOG,TESTOG,ONLTOG;                    00066200
  BEGIN                                                                      00066400
    DI:=L;                                                                   00066500
    DS:=12 LIT"LOGANL/MAINT";                                                00066550
    DS:=12 LIT"      RUN   ";                                                00066600
    SI:=LOC JULIAN; SI:=SI+3; DS:=5 CHR;                                     00066650
    DI:=DI+2; SI:=SI+2; DS:=6 CHR;                                           00066700
    DS:=7 LIT"DAY,   "; DS:=8 CHR;                                           00066800
    DI:=DI+3; SI:=SI+3; DS:=5 CHR;                                           00066900
    DS:=9 LIT"   WITH  ";                                                    00067000
    LISTOG(DS:=14 LIT"CHRONOLOGICAL,");                                      00067100
    SUMTOG(DS:=8 LIT"SUMMARY,");                                             00067200
    TESTOG(DS:=13 LIT"REALTIMETAPE,");                                       00067300
    ONLTOG(DS:=11 LIT"CONFIDENCE,");                                         00067400
    DI:=DI-1; DS:=LIT".";                                                    00067500
  END BOJMESS;                                                               00067600
%                                                                            00067700
STREAM PROCEDURE BLANK(L);                                                   00069000
  BEGIN                                                                      00069100
    DI:=L; DS:=8 LIT" "; SI:=L; DS:=14 WDS;                                  00069200
  END BLANK;                                                                 00069300
%                                                                            00069400
STREAM PROCEDURE UNDERLINE(L);                                               00070000
  BEGIN                                                                      00071000
    SI:=L; DI:=L;                                                            00072000
    2(60(IF SC#" " THEN DS:=LIT"-" ELSE DI:=DI+1; SI:=SI+1));                00073000
  END UNDERLINE;                                                             00074000
%                                                                            00074100
STREAM PROCEDURE ZEROUT(N,A); VALUE N;                                       00075000
  BEGIN LOCAL NDIV64;                                                        00076000
    SI:=LOC N; DI:=LOC NDIV64; SI:=SI+6; DI:=DI+7; DS:=CHR;                  00077000
    DI:=A; DS:=8 LIT"0";  SI:=A;                                             00078000
    NDIV64(DS:=32 WDS; DS:=32 WDS); DS:=N WDS;                               00079000
  END ZEROUT;                                                                00080000
%                                                                            00080100
STREAM PROCEDURE MOVE(N,HERE,THERE); VALUE N;                                00081000
  BEGIN                                                                      00082000
    SI:=HERE; DI:=THERE; DS:=N WDS;                                          00083000
  END MOVE;                                                                  00084000
```

```
%                                                                    00084050
REAL STREAM PROCEDURE SNOOP(FILEN);                                  00084100
  BEGIN                                                              00084200
    SI:=FILEN; DI:=LOC SNOOP; DS:=WDS;                               00084300
  END SNOOP;                                                         00084400
%                                                                    00084450
REAL STREAM PROCEDURE GETEUSPEED(EU,EUSPEED); VALUE EU,EUSPEED;      00084500
  BEGIN                                                              00084600
    DI:=LOC GETEUSPEED; DI:=DI+7, SKIP 4 DB;                         00084700
    SI:=LOC EUSPEED; SKIP 8B; EU(SKIP 2 SB); 2(BIT);                 00084800
  END GETEUSPEED;                                                    00084900
%                                                                    00085000
REAL STREAM PROCEDURE OCTV(DECI); VALUE DECI;                        00086000
  BEGIN                                                              00087000
    DI:=LOC DECI; DS:=5 RESET; DI:=LOC OCTV; SI:=LOC DECI; DS:=8 OCT; 00088000
  END OCTV;                                                          00089000
%                                                                    00089100
REAL STREAM PROCEDURE OCTCONV(N,DECI); VALUE N,DECI;                 00090000
  BEGIN                                                              00091000
    SI:=LOC DECI; SI:=SI+N; DI:=LOC OCTCONV; 8(OCTADE);              00092000
  END OCTCONV;                                                       00093000
%                                                                    00093100
REAL STREAM PROCEDURE DECM(OCTV); VALUE OCTV;                        00094000
  BEGIN                                                              00095000
    SI:=LOC OCTV; DI:=LOC DECM; DS:=8 DEC;                           00096000
  END DECM;                                                          00097000
%                                                                    00097100
REAL STREAM PROCEDURE DECL(OCTV); VALUE OCTV;                        00098000
  BEGIN                                                              00099000
    SI:=LOC OCTV; DI:=LOC DECL; DS:=8 DEC; DI:=DI-7; DS:=7 FILL;     00100000
  END DECL;                                                          00101000
%                                                                    00102000
REAL PROCEDURE ATIME(XCLOCK); VALUE XCLOCK; REAL XCLOCK;             00103000
  BEGIN INTEGER HR,MIN;                                              00104000
    HR:= XCLOCK DIV 216000;                                          00105000
    MIN:= (XCLOCK DIV 3600) MOD 60;                                  00106000
    ATIME:= DECM(MIN) & 13[30:42:6] & DECM(HR)[18:36:12];            00107000
  END ATIME;                                                         00108000
%                                                                    00109000
REAL PROCEDURE DAY(JULIAN); VALUE JULIAN; REAL JULIAN;               00110000
  BEGIN                                                              00111000
    INTEGER D,H,M,Q,P,Y;                                             00112000
    LABEL OWT,EXIT;                                                  00113000
    IF JULIAN=0 THEN BEGIN ACTDATE:=0; GO EXIT; END;                 00114000
    D:=JULIAN.[30:6]×100+JULIAN.[36:6]×10+JULIAN.[42:6];             00115000
    Y:=JULIAN.[18:6]×10 +JULIAN.[24:6];                              00116000
    FOR H:=31, IF Y MOD 4 = 0 THEN 29 ELSE 28,                       00117000
          31,30,31,30,31,31,30,31,30 DO                              00118000
          IF D LEQ H THEN GO OWT ELSE                                00119000
          BEGIN D:=D-H; M:=M+1; END;                                 00120000
OWT:                                                                 00121000
    IF M<2 THEN BEGIN Q:=M+11; P:=Y-1 END                            00122000
           ELSE BEGIN Q:=M- 1; P:=Y   END;                           00123000
    M:=M+1;                                                          00124000
    ACTDATE:= JULIAN.[18:12] & 49[30:42:6] & DECM(D)[18:36:12]       00125000
                             & 49[12:42:6] & DECM(M)[1 :37:11];      00126000
    IF JULIAN < 0 THEN                                               00127000
    DAY:= ACTDATE ELSE                                               00128000
    DAY:= DAYS[((Q×26-2) DIV 10+D+P+P.[36:10]+1) MOD 7];             00129000
EXIT:                                                                00130000
```

```
  END DAY;                                                                  00131000
%                                                                           00132000
BOOLEAN STREAM PROCEDURE TESTBIT(B,I); VALUE I;                             00133000
  BEGIN                                                                     00134000
    SI:=B; SI:=SI+36; SKIP 2 SB;                                            00135000
    SKIP I SB; IF SB THEN TALLY:=1;                                         00136000
    TESTBIT:= TALLY;                                                        00137000
  END TESTBIT;                                                              00138000
%                                                                           00139000
INTEGER STREAM PROCEDURE ANALYZEDESC(TYPE,TINU,RESULT,MSGBUFFER);           00140000
  VALUE TYPE,TINU;                                                          00141000
  BEGIN LOCAL R;                                                            00142000
    LABEL CARD,LINE,TAPE,DRUM,DISK,SPO,                                     00143000
          PUNCH,PREAD,D19,XIT;                                              00144000
    DI:=MSGBUFFER; DS:=8 LIT" ";                                            00144100
    SI:=MSGBUFFER; DS:=39 WDS;                                              00144200
    DI:=MSGBUFFER;                                                          00145000
    SI:=RESULT; SI:=SI+4; IF SB THEN TALLY:=1;                              00146000
    R:=TALLY; TALLY:=0; SKIP 2 SB;                                          00146100
    IF SB THEN                                                              00147000
    BEGIN                                                                   00148000
        DS:=24 LIT"D22 - CORE ADDRESS ERROR";                               00149000
        DS:=11 LIT" ADDRESSS= @";                                           00150000
        SKIP 7 SB; 5(OCTADE);                                               00152000
        SI:=SI-4; SKIP 2 SB;                                                00154000
        TALLY:=TALLY+1;                                                     00155000
    END;                                                                    00156000
    SKIP 4 SB;                                                              00157000
    IF SB THEN                                                              00158000
    BEGIN                                                                   00159000
        SI:=LOC TINU; SI:=SI+5;                                             00160000
        DS:=6 LIT"D18 - "; DS:=3 CHR;                                       00161000
        DS:=10 LIT" NOT READY"; DI:=DI+21;                                  00162000
        SI:=RESULT; SI:=SI+5;                                               00163000
        TALLY:=TALLY+1;                                                     00164000
    END;                                                                    00165000
    SKIP SB;                                                                00166000
    IF SB THEN                                                              00167000
    BEGIN                                                                   00168000
        DS:=18 LIT"D17 - DESC. PARITY"; DI:=DI+22;                          00169000
        TALLY:=TALLY+1;                                                     00170000
    END;                                                                    00171000
    SKIP SB;                                                                00172000
    IF SB THEN                                                              00173000
    BEGIN                                                                   00174000
        SI:=LOC TINU; SI:=SI+5;                                             00175000
        DS:=6 LIT"D16 - "; DS:=3 CHR;                                       00176000
        DS:=5 LIT" BUSY"; DI:=DI+26;                                        00177000
        TALLY:=TALLY+1;                                                     00178000
    END;                                                                    00179000
    SI:=RESULT; SI:=SI+4; SKIP 3 SB;                                        00180000
    CI:=CI+TYPE;                                                            00181000
    GO TO XIT;                                                              00182000
    GO TO CARD;                                                             00183000
    GO TO LINE;                                                             00184000
    GO TO TAPE;                                                             00185000
    GO TO DRUM;                                                             00186000
    GO TO DISK;                                                             00187000
    GO TO SPO;                                                              00188000
    GO TO PUNCH;                                                            00189000
```

```
        GO TO XIT;                                                          00190000
        GO TO SPO;      % PPUNCH...                                         00191000
        GO TO PREAD;                                                        00192000
        GO TO SPO;      % CCOM.....                                         00193000
    CARD:                                                                   00194000
        SKIP SB;                                                            00195000
        IF SB THEN                                                          00196000
        BEGIN                                                               00197000
            DS:=16 LIT"D20 - READ CHECK"; DI:=DI+24;                        00198000
            TALLY:=TALLY+1;                                                 00199000
        END;                                                                00200000
        SKIP SB;                                                            00201000
        IF SB THEN                                                          00202000
        BEGIN                                                               00203000
            DS:=20 LIT"D19 - VALIDITY CHECK";                               00204000
            TALLY:= TALLY+1;                                                00205000
        END;                                                                00206000
        GO TO XIT;                                                          00207000
    LINE:                                                                   00208000
        SKIP SB;                                                            00209000
        IF SB THEN                                                          00210000
        BEGIN                                                               00211000
            DS:=17 LIT"D20 - PRINT CHECK"; DI:=DI+23;                       00212000
            TALLY:=TALLY+1;                                                 00213000
        END;                                                                00214000
        GO TO D19;                                                          00216000
    TAPE:                                                                   00217000
        SKIP SB;                                                            00218000
        IF SB THEN                                                          00219000
        BEGIN                                                               00220000
            DS:=6 LIT"D20 - ";                                              00222000
            DS:=5 LIT"WRITE"; R(DI:=DI-5; DS:=5 LIT"READ ");                00223000
            DS:=7 LIT" PARITY"; DI:=DI+22;                                  00224000
            TALLY:=TALLY+1;                                                 00226000
    END;                                                                    00227000
    SKIP SB;                                                                00228000
    IF SB THEN ELSE GO TO XIT; % SAVES PRT LOCATION                         00229000
    BEGIN                                                                   00230000
        SI:=RESULT; SKIP 2 SB;                                              00231000
        IF SB THEN ELSE  BEGIN SI:=SI+3; SKIP 4 SB; GO TO D19 END;          00232000
        BEGIN                   % MOD III I/O"S                             00233000
            SKIP 9 SB;                                                      00234000
            IF SB THEN                                                      00235000
            BEGIN                                                           00236000
                DS:=20 LIT"D37,D19 - MEM PARITY"; DI:=DI+20;                00237000
                TALLY:=TALLY+1;                                             00238000
            END;                                                            00239000
            SKIP SB;                                                        00240000
            IF SB THEN                                                      00241000
            BEGIN                                                           00242000
                DS:=24 LIT"D36,D19 - BLANK TAPE ON ";                       00244000
                DS:=5 LIT"WRITE"; R(DI:=DI-5; DS:=5 LIT"READ ");            00245000
                DI:=DI+11;                                                  00246000
                TALLY:=TALLY+1;                                             00248000
            END;                                                            00249000
            SKIP SB;                                                        00250000
            IF SB THEN                                                      00251000
            BEGIN                                                           00252000
                DS:=18 LIT"D35,D19 - TAPE BOT"; DI:=DI+22;                  00253000
                TALLY:=TALLY+1;                                             00254000
```

```
                END;                                                                    00255000
                SKIP SB;                                                                00256000
                IF SB THEN                                                              00257000
                BEGIN                                                                   00258000
                     DS:=18 LIT"D34,D19 - TAPE EOT";                                    00259000
                     TALLY:=TALLY+1;                                                    00260000
                END;                                                                    00261000
                DI:=RESULT; DI:=DI+4; SKIP 3 DB;                                        00262000
                SI:=SI-1;                                                               00263000
                IF SB THEN DS:=SET ELSE DS:=RESET;                                      00264000
                SKIP DB;                                                                00265000
                SI:=SI-1; SKIP 5 SB;                                                    00266000
                IF SB THEN ELSE DS:=RESET;                                              00267000
          END; END;                                                                     00271000
          GO TO XIT;                                                                    00272000
     DRUM:                                                                              00273000
          IF SB THEN                                                                    00274000
          BEGIN                                                                         00275000
               DS:=27 LIT"D21 - DRUM CHANNEL LOCK OUT"; DI:=DI+13;                      00276000
               TALLY:=TALLY+1;                                                          00277000
          END;                                                                          00278000
          SKIP SB;                                                                      00279000
          IF SB THEN                                                                    00280000
          BEGIN                                                                         00281000
               DS:=17 LIT"D20 - MEMORY RACE"; DI:=DI+23;                                00282000
               TALLY:=TALLY+1;                                                          00283000
          END;                                                                          00284000
          SKIP SB;                                                                      00285000
          IF SB THEN                                                                    00286000
          BEGIN                                                                         00287000
               R(DS:=17 LIT"D19 - READ PARITY"; TALLY:=TALLY+1; JUMP OUT XIT);          00288000
               SI:=SI-1; SKIP 4 SB;                                                     00289000
               GO TO D19;                                                               00290000
          END;                                                                          00296000
          GO TO XIT;                                                                    00297000
     DISK:                                                                              00298000
          SI:=SI-1; SKIP SB;                                                            00298100
          IF SB THEN                                                                    00298120
          BEGIN                                                                         00298140
               DS:=16 LIT"D23 - READ CHECK"; DI:=DI+24;                                 00298160
               TALLY:=TALLY+1;                                                          00298180
          END;                                                                          00298200
          SKIP 2 SB;                                                                    00298220
          IF SB THEN                                                                    00299000
          BEGIN                                                                         00300000
               DS:=18 LIT"D21 - EU NOT READY"; DI:=DI+22;                               00301000
               TALLY:=TALLY+1;                                                          00302000
          END;                                                                          00303000
          SKIP SB;                                                                      00304000
          IF SB THEN                                                                    00305000
          BEGIN                                                                         00306000
               DS:=17 LIT"D20 - WRITE LOCK ";                                           00307000
               R(DI:=DI-11; DS:=11 LIT"READ PARITY"); DI:=DI+23;                        00307100
               TALLY:=TALLY+1;                                                          00308000
          END;                                                                          00309000
          SKIP SB;                                                                      00310000
          IF SB THEN                                                                    00311000
          BEGIN                                                                         00312000
               DS:=17 LIT"D19 - DATA PARITY";                                           00313000
               TALLY:=TALLY+1;                                                          00314000
```

```
        END;                                                                00315000
        GO TO XIT;                                                          00316000
    PUNCH:                                                                  00317000
        SKIP SB;                                                            00318000
        IF SB THEN                                                          00319000
        BEGIN                                                               00320000
            DS:=17 LIT"D20 - PUNCH CHECK"; DI:=DI+23;                       00321000
            TALLY:=TALLY+1;                                                 00322000
        END;                                                                00323000
        GO TO D19;                                                          00325000
    PREAD:                                                                  00326000
        SKIP 2 SB;                                                          00327000
        IF SB THEN                                                          00328000
        BEGIN                                                               00329000
            DS:=17 LIT"D19 - READ PARITY";                                  00330000
            TALLY:=TALLY+1;                                                 00331000
        END;                                                                00332000
        GO TO XIT;                                                          00333000
    SPU:                                                                    00334000
        SKIP SB;                                                            00336000
    D19:                                                                    00337000
        SKIP SB;                                                            00338000
        IF SB THEN                                                          00339000
        BEGIN                                                               00340000
            DS:=16 LIT"D19 - MEM PARITY";                                   00341000
            TALLY:=TALLY+1;                                                 00342000
        END;                                                                00343000
    XIT:                                                                    00344000
        ANALYZEDESC:=TALLY;                                                 00345000
     END ANALYZEDESC;                                                       00346000
    %                                                                       00347000
    PROCEDURE BREAKDOWNADDRESS(FA,ARAY);                                    00348000
      VALUE FA; REAL FA; INTEGER ARRAY ARAY[0];                             00349000
      BEGIN                                                                 00350000
       DEFINE                                                               00351000
          FILEADDR=ARAY[0]#,                                                00352000
          EU      =ARAY[1]#,                                                00353000
          SU      =ARAY[2]#,                                                00354000
          SET     =ARAY[3]#,                                                00355000
          FACE    =ARAY[4]#,                                                00356000
          ZONE    =ARAY[5]#,                                                00357000
          TRACK   =ARAY[6]#,                                                00358000
          SEG     =ARAY[7]#,                                                00359000
          HEAD    =ARAY[8]#,                                                00360000
          PIN     =ARAY[9]#,                                                00361000
          ERROR(ERROR1)=BEGIN ARAY[9]:=-ERROR1; GO EXIT; END#;              00362000
    %                                                                       00362100
      INTEGER C,TR,DKTYPE;                                                  00363000
      LABEL HS1,HS2,HS3,HS4,EXIT;                                           00364000
      SWITCH HSW:=HS1,HS2,HS3,HS4;                                          00365000
      BOOLEAN EVEN;                                                         00366000
    %                                                                       00366100
      INTEGER PROCEDURE ZNE(FA,DKTYPE);                                     00367000
       VALUE FA,DKTYPE; INTEGER FA,DKTYPE;                                  00368000
       ZNE:= IF FA:=(FA MOD 100) < (IF DKTYPE=2 THEN 25 ELSE 24) THEN 0     00369000
             ELSE IF FA < (IF DKTYPE=2 THEN 58 ELSE 56) THEN 1 ELSE 2;      00370000
    ;                                                                       00371000
       FILEADDR:= BCTV(FA);                                                 00372000
       IF (EU:=OCTV(FA.[5:7])) > 19 THEN ERROR(1);                          00373000
       IF (DKTYPE:=GETEUSPEED(EU,EUSPEED)) = 0 THEN ERROR(2);               00374000
```

```
        IF (SET:=OCTV(FA.[12:12])) > 20×DKTYPE-1 THEN ERROR(3);                  00375000
        IF (TR:=TRACK:=OCTV(FA.[24:12])) > 99 THEN ERROR(4);                     00376000
        IF (SEG:=ABS(OCTV(FA.[36:12]))) > 99 THEN ERROR(5);                      00377000
        SU:= SET DIV (4×DKTYPE) +1;                                              00378000
        SET:= (SET MOD 4) +1;                                                    00379000
        FACE:=IF (REAL(TR<50) + REAL(SET MOD 2=1)) MOD 2=0                       00380000
              THEN "CW " ELSE "CCW";                                             00381000
        ZONE:=ZNE(FILEADDR,DKTYPE)+1;                                            00382000
        EVEN:=TR MOD 2=0;                                                        00383000
        C:= IF TR LEQ 24 THEN IF EVEN THEN 1 ELSE 0 ELSE                         00384000
            IF TR LEQ 49 THEN IF EVEN THEN 3 ELSE 2 ELSE                         00385000
            IF TR LEQ 74 THEN IF EVEN THEN 1 ELSE 0 ELSE                         00386000
            IF EVEN THEN 3 ELSE 2;                                               00387000
        TR:= TR MOD 50;                                                          00388000
        GO TO HSW[C+1];                                                          00389000
    HS1:   %C=0                                                                  00390000
        PIN:=TR DIV 2+5;                                                         00391000
        HEAD:=IF ZONE = 1 THEN 1 ELSE                                            00392000
              IF ZONE = 2 THEN 5 ELSE 11;                                        00393000
        GO EXIT;                                                                 00394000
    HS2:   %C=1                                                                  00395000
        PIN:=TR DIV 2 + 4;                                                       00396000
        HEAD:=IF ZONE=1 THEN 2 ELSE                                              00397000
              IF ZONE=2 THEN 8 ELSE 12;                                          00398000
        GO EXIT;                                                                 00399000
    HS3:   %C=2                                                                  00400000
        PIN:=((TR-24) DIV 2)+4;                                                  00401000
        HEAD:=IF ZONE=1 THEN 3 ELSE                                              00402000
              IF ZONE=2 THEN 7 ELSE 13;                                          00403000
        GO EXIT;                                                                 00404000
    HS4:   %C=3                                                                  00405000
        PIN:=((TR-26) DIV 2)+5;                                                  00406000
        HEAD:=IF ZONE=1 THEN 4 ELSE                                              00407000
              IF ZONE=2 THEN 6 ELSE 14;                                          00408000
        GO EXIT;                                                                 00409000
    EXIT:                                                                        00410000
      END BREAKDOWNADDRESS;                                                      00411000
    %                                                                            00414000
    PROCEDURE PERIPHANDLER;                                                      00415000
      BEGIN REAL IO,NEUM,ENTRY;                                                  00416000
        LABEL READOBJ,READFIL,NOFILE,SEEK,EXIT;                                  00417000
        DEFINE FILEADDR=A[0]#,                                                   00417100
               EU       =A[1]#,                                                  00417200
               SU       =A[2]#,                                                  00417300
               SET      =A[3]#,                                                  00417400
               FACE     =A[4]#,                                                  00417500
               ZONE     =A[5]#,                                                  00417600
               TRACK    =A[6]#,                                                  00417700
               SEG      =A[7]#,                                                  00417800
               HEAD     =A[8]#,                                                  00417900
               PIN      =A[9]#;                                                  00418000
        FORMAT                                                                   00418500
            DETAIL(A5,X3,A3," (",A1,")",X2,"MIX=",I3,X3,"CHANNEL=",I2),          00419000
            DAF(X40,"DISK ADDRESS =  ",I2,A6),                                   00420000
            ADDRESSERRORF(X40,"DISK ADDRESS ERROR"),                             00420100
            EUF(X40,"EU",I2),                                                    00420200
            SUF(X40,"STORAGE UNIT",I3,X2,"I",X2,"ZONE",I2,X6,"I",                00420300
                X2,"HEAD",I3),                                                   00420400
            DISKF(X40,"DISK",I2,X11,":",X2,"TRACK",I3,X4,":",X2,                 00420500
                  "PIN ",I2," (CENTER TAP)"),                                    00420600
```

```
  FACEF(X40,A3," FACE",X9,":",X2,"SEGMENT ",I2,X2,":"),                    00420700
  PHYREEL(X40,"PHYSICAL REEL=  ",I5),                                      00421000
  IODESC(X40,"I/O DESC.    =",X2,O,X1,O,                                   00422000
       *(X15,"OBJ JOB NAME  =",X2,A1,A6,"/",A1,A6)),                       00423000
  REDESC(X40,"RESULT DESC, =",X2,O,X1,O,                                   00424000
       *(X15,"FILE NAME     =",X2,A1,A6,"/",A1,A6)),                       00425000
  RETRIES(X40,"RETRIES      =  ",I5),                                      00426000
  TAPENAME(X88,"TAPE NAME      =  ",A1,A6,"/",A1,A6),                      00427000
  TAPEREEL(X88,"REEL           =  ",I5),                                   00428000
  TAPEDATE(X88,"CREATION DATE =  ",A5,X2,O),                               00429000
  TAPECYCL(X88,"CYCLE          =  ",I5),                                   00430000
  TAPETRAN(X88,"TRANSACTION    =  ",I5);                                   00431000
%                                                                          00432000
MIX := B[0].[20:5];                                                        00433000
IF (FNUM:=B[0].[9:9])>MAXFNUM THEN FNUM:=MAXFNUM;                          00434000
LUN := UNIT[B[3].[3:5]x2+IO:=B[3].[24:1]];                                 00435000
IF TYPE=3 THEN IF B[9]=1023 THEN                                           00436000
BEGIN IF LUN<16 THEN RETRY[LUN]:=RETRY[LUN]+B[8];                          00438000
      GO EXIT;                                                             00439000
END;                                                                       00440000
IF L[STOG THEN                                                             00441000
BEGIN                                                                      00442000
IF TYPE=5 THEN WRITE(L[*],DAF,B[2].[5:7],B[2]) ELSE                        00443000
IF TYPE=3 THEN WRITE(L[*],PHYREEL,B[7].[30:18]) ELSE BLANK(L);             00444000
WRITE(A[*],DETAIL                                                          00446000
          ,ATIME(B[1].[3:24])                                              00447000
          ,NEUM:=IF LUN=3, THEN UNK[TYPE] ELSE TINU[LUN]                   00448000
          ,IF IO=1 THEN 41 ELSE 54                                         00449000
          ,MIX                                                             00450000
          ,B[0].[18:2]+1                                                   00451000
         );                                                                00452000
MOVE(5,A,L);                                                               00453000
WRITESINGLE;                                                               00454000
IF TYPE=5 THEN                                                             00455000
BEGIN WRITE(L[*],RETRIES,IF (I:=B[2].[1:4])=0 THEN 10 ELSE I);             00456000
      IF I=0 THEN MOVE(2,MESS[6],L[8]) ELSE MOVE(2,MESS[4],L[8]);          00458000
      WRITESINGLE;                                                         00460000
END ELSE                                                                   00461000
IF TYPE=3 THEN                                                             00462000
BEGIN WRITE(L[*],RETRIES,(B[8] + RETRY[LUN]) DIV 2);                       00463000
      IF (I:=B[9]) = 0 THEN MOVE(2,MESS[4],L[8]) ELSE                      00465000
      IF I=8  THEN MOVE(2,MESS[8],L[8]) ELSE                               00466000
      IF I=16 THEN MOVE(2,MESS[6],L[8]) ELSE                               00467000
      IF I=32 THEN MOVE(2,MESS[10+2xIO],L[8]);                             00468000
      WRITESINGLE;                                                         00469000
END;                                                                       00470000
BLANK(L); WRITEDOUBLE;                                                     00471000
END;                                                                       00472000
IF MIX≠0 THEN                                %OBJ JOB RELATED              00473000
IF TYPE=5 THEN                               %DISK LOOK AHEAD              00474000
IF JAR[MIX,0]=0 THEN                         %NO JOB ENTRY                 00475000
BEGIN                                                                      00476000
  ENTRY:= RCNT;                                                            00477000
READOBJ:                                                                   00478000
  READ(LOG,5,A[*])[SEEK];                                                  00479000
  ENTRY:= ENTRY+1;                                                         00480000
  IF (T1:= A[0].[3:6])≠12 THEN          %CHECK FOR H/L                     00481000
  IF T1=63 OR T1=15 THEN                % END OF LOG OR HALT/LOAD           00482000
  BEGIN                                                                    00483000
       JAR[MIX,0]:= 1;                                                     00484000
```

```
                MOVE(2,MESS[0+REAL(T1=63)*2],JAR[MIX,1]);                    00485000
                IF FNUM=0 THEN GO SEEK ELSE GO NOFILE;                       00486000
            END ELSE                                                         00487000
            BEGIN SPACE(LOG,I:=A[0].[39:9])[SEEK];                           00488000
                  ENTRY:= ENTRY+I;                                           00489000
                  GO READOBJ;                                                00490000
            END ELSE                                                         00491000
            IF A[0].[20:5]≠MIX THEN GO READOBJ;                              00492000
            A[2].[3:15]:= ENTRY;                                             00493000
            MOVE(3,A[2],JAR[MIX,0]);                                         00494000
            IF FNUM=0 THEN GO SEEK;                                          00495000
READFIL:                                                                     00496000
            READ(LOG,5,A[*])[SEEK];                                          00497000
            ENTRY:= ENTRY+1;                                                 00498000
            IF (T1:=A[0].[3:6])≠13 THEN          %CHECK FOR H/L              00499000
            IF T1=63 OR T1=15 THEN               % END OF LOG OR HALT/LOAD    00500000
            BEGIN                                                            00501000
NOFILE:           JAR[MIX,FNUM*3]:= 1;                                       00502000
                  MOVE(2,MESS[0+REAL(T1=63)*2],JAR[MIX,FNUM*3+1]);           00503000
                  GO SEEK;                                                   00504000
            END ELSE                                                         00505000
            BEGIN SPACE(LOG,I:=A[0].[39:9])[SEEK];                           00506000
                  ENTRY:= ENTRY+I;                                           00507000
                  GO READFIL;                                                00508000
            END ELSE                                                         00509000
            IF A[0].[20:5]≠MIX THEN GO READFIL ELSE                          00510000
            IF A[0].[9:9]≠FNUM THEN GO READFIL;                              00511000
            A[2].[3:15]:= ENTRY;                                             00512000
            MOVE(3,A[2],JAR[MIX,FNUM*3]);                                    00513000
SEEK:       READ SEEK(LOG[RCNT+1]);                                          00514000
          END ELSE                                                           00515000
          IF FNUM≠0 THEN                                                     00516000
          IF JAR[MIX,FNUM*3]=0 THEN                                          00517000
          BEGIN ENTRY:= RCNT;                                                00518000
                GO READFIL;                                                  00519000
          END;                                                               00520000
          IF LISTOG THEN                                                     00521000
          BEGIN                                                              00522000
          IF MIX=0 THEN                                                      00523000
          BEGIN WRITE(L[*],IODESC                                            00524000
                         ,OCTCONV(0,B[3])                                    00525000
                         ,OCTCONV(4,B[3])                                    00526000
                         ,"1");                                              00527000
                WRITESINGLE;                                                 00528000
          END ELSE                                                           00529000
          BEGIN                                                              00530000
               WRITE(L[*],IODESC                                             00531000
                         ,OCTCONV(0,B[3])                                    00532000
                         ,OCTCONV(4,B[3])                                    00533000
                         ,1                                                  00534000
                         ,(I:=JAR[MIX,1]).[6:6],I                            00535000
                         ,(I:=JAR[MIX,2]).[6:6],I                            00536000
                        );                                                   00537000
               WRITESINGLE;                                                  00538000
          END;                                                               00539000
          IF FNUM=0 THEN                                                     00540000
          BEGIN WRITE(L[*],REDESC                                            00541000
                         ,OCTCONV(0,B[4])                                    00542000
                         ,OCTCONV(4,B[4])                                    00543000
                         ,"1");                                              00544000
```

```
        WRITEDOUBLE;                                                    00545000
    END ELSE                                                            00546000
    BEGIN WRITE(L[*],REDESC                                             00547000
                    ,OCTCONV(0,B[4])                                    00548000
                    ,OCTCONV(4,B[4])                                    00549000
                    ,1                                                  00550000
                    ,(I:=JAR[MIX,FNUM*3+1]),[6:6],I                     00551000
                    ,(I:=JAR[MIX,FNUM*3+2]),[6:6],I                     00552000
                   );                                                   00553000
        WRITEDOUBLE;                                                    00554000
    END;                                                                00555000
    IF (I:=(ANALYZEDESC(TYPE,NEUM,B[4],JUNKBUFFER)-1)*5) GEQ 0 THEN     00556000
    BEGIN                                                               00557000
     BLANK(L);                                                          00558000
     FOR J:=0 STEP 5 UNTIL I DO                                         00559000
     BEGIN                                                              00560000
     MOVE(5,JUNKBUFFER[J],L[7]);                                        00561000
     WRITESINGLE;                                                       00562000
    END;END;                                                            00563000
    BLANK(L); WRITEDOUBLE;                                              00564000
    END ELSE IF TYPE=3 THEN I:=ANALYZEDESC(3,0,B[4],JUNKBUFFER);        00565000
    IF SUMTOG THEN                                                      00565100
    BEGIN                                                               00565200
      MOVE(7,B,JUNKBUFFER);                                             00565300
      MOVE(2,JAR[MIX,1],JUNKBUFFER[8]);                                 00565400
      IF TYPE#3 THEN MOVE(2,JAR[MIX,FNUM*3+1],JUNKBUFFER[5]);           00565500
      JUNKBUFFER[1]:= JUNKBUFFER[1] & DCIV(CDATE)[27:27:21];            00565600
      JUNKBUFFER[7]:= (B[0],[18:2]+1) & LUN[1:43:5];                    00565700
      SUMFILETOG:= TRUE;                                                00565800
    END;                                                                00565900
    IF TYPE=5 THEN                                                      00566000
    BEGIN                                                               00567000
      BREAKDOWNADDRESS(B[2],A);                                         00568000
      IF LISTOG THEN                                                    00569000
      BEGIN IF A[9]<0 THEN WRITE(L[*],ADDRESSERRORF) ELSE               00569100
            BEGIN WRITE(L[*],EUF,EU); WRITESINGLE;                      00569200
                  WRITE(L[*],SUF,SU,ZONE,HEAD); WRITESINGLE;            00569300
                  WRITE(L[*],DISKF,SET,TRACK,PIN); WRITESINGLE;         00569400
                  WRITE(L[*],FACEF,FACE,SEG);                           00569500
            END;                                                        00569600
            WRITEDOUBLE;                                                00569700
      END;                                                              00569800
      IF SUMTOG THEN                                                    00570000
      BEGIN                                                             00571000
        JUNKBUFFER[7]:= JUNKBUFFER[7] & REAL(A[9]<0)[7:47:1] &          00572000
                       FILEADDR[8:23:25] & B[3][33:27:6] &              00573000
                      (IF (I:=B[2],[1:4])=0 THEN 10 ELSE I)[39:42:6];   00574000
      END;                                                              00576000
    END ELSE                                                            00579000
    IF TYPE=3 THEN                                                      00580000
    BEGIN IF LISTOG THEN                                                00581000
        BEGIN                                                           00582000
          WRITE(L[*],TAPENAME                                           00583000
                    ,B[5],[6:6],B[5]                                    00584000
                    ,B[6],[6:6],B[6]);        WRITESINGLE;              00585000
          WRITE(L[*],TAPEREEL,B[2],[14:10]); WRITESINGLE;               00586000
          WRITE(L[*],TAPEDATE                                           00587000
                    ,IF (I:=DECM(B[2],[24:17]))=0                       00588000
                     ANB IQ=0 THEN I:=CDATE ELSE I                      00589000
                    ,DAY(-I)); WRITESINGLE;                             00590000
```

```
                WRITE(L[*],TAPECYCL,B[2].[41:7]); WRITESINGLE;          00591000
                WRITE(L[*],TAPETRAN,B[1].[27:21]); WRITEDOUBLE;         00592000
              END;                                                      00593000
              IF SUMTOG THEN                                            00594000
              BEGIN                                                     00595000
                RETRY[LUN]:= 0;                                         00596000
                JUNKBUFFER[0]:= JUNKBUFFER[0] & B[1][27:27:21];         00597000
                JUNKBUFFER[7]:= JUNKBUFFER[7] & B[7][15:30:18] &        00598000
                                LUN[33:42:6] & B[9][39:42:6];           00599000
              END;                                                      00601000
          END ELSE                                                      00602000
          IF TYPE=4 THEN DRUMTOG:= TRUE;                                00603000
          IF LISTOG THEN BEGIN BLANK(L); WRITESPACES(4); END;           00604000
          IF SUMTOG THEN WRITE(TEMP,10,JUNKBUFFER[*]);                  00605000
EXIT:                                                                   00606000
  END PERIPHANDLER;                                                     00607000
%                                                                       00611000
PROCEDURE GROUPTOTALS(J,COUNT);                                         00612000
  VALUE J; REAL J,COUNT;                                                00613000
  BEGIN                                                                 00614000
    REAL IO,KEY,S,UERR,UTITLE1,UTITLE2,LUNMIN,LUNMAX,LASTL,LASTK;       00615000
    ARRAY TOTERR[0:13];                                                 00615100
    LABEL LOOP,COMPARE,ADDUP,WRITEIT,EXIT;                              00616000
    SWITCH FORMAT SWFORM:=                                              00617000
%MAG TAPE                                                               00618000
(X4,"UNIT",X7,"TOTALS",X5,"MEM ADDRESS",X4,"BLANK TAPE",X3,"TAPE PARITY"00619000
        ,X4,"MEM PARITY",X5,"NOT READY",X3,"DESC PARITY",X7,"BUSY"),    00620000
%DISK                                                                   00621000
(X4,"UNIT",X7,"TOTALS",X5,"MEM ADDRESS  EU NOT READY",X3,"READ PARITY", 00622000
        X3,"DATA PARITY",X2,"CU NOT READY",X3,"DESC PARITY",X7,"BUSY"),00623000
%DRUM                                                                   00624000
(X4,"UNIT",X7,"TOTALS",X5,"MEM ADDRESS",X4,"WRITE LOCK",X3,"MEMORY RACE"00625000
        ,X4,"MEM PARITY",X5,"NOT READY",X3,"DESC PARITY",X7,"BUSY"),    00626000
%PRINTER                                                                00627000
(X4,"UNIT",X7,"TOTALS",X5,"MEM ADDRESS",X17,"PRINT CHECK",              00628000
        X4,"MEM PARITY",X5,"NOT READY",X3,"DESC PARITY",X7,"BUSY"),     00629000
%CARD PUNCH                                                             00630000
(X4,"UNIT",X7,"TOTALS",X5,"MEM ADDRESS",X17,"PUNCH CHECK",              00631000
        X4,"MEM PARITY",X5,"NOT READY",X3,"DESC PARITY",X7,"BUSY"),     00632000
%CARD READ                                                              00633000
(X4,"UNIT",X7,"TOTALS",X5,"MEM ADDRESS",X18,"READ CHECK",               00634000
        X4,"VALIDTY CK",X5,"NOT READY",X3,"DESC PARITY",X7,"BUSY"),     00635000
%SPO                                                                    00636000
(X4,"UNIT",X7,"TOTALS",X5,"MEM ADDRESS",X28,                            00637000
        X4,"SPO PARITY",X5,"NOT READY",X3,"DESC PARITY",X7,"BUSY"),     00638000
%PAPER TAPE                                                             00639000
(X4,"UNIT",X7,"TOTALS",X5,"MEM ADDRESS",X28,                            00640000
        X3,"TAPE PARITY",X5,"NOT READY",X3,"DESC PARITY",X7,"BUSY"),    00641000
%DCOM                                                                   00642000
(X4,"UNIT",X7,"TOTALS",X5,"MEM OVRFLOW",X28,                            00643000
        X4,"MEM PARITY",X1,"DTC NOT READY",X3,"DESC PARITY",X3,"BUSY");00644000
%                                                                       00644100
      FORMAT                                                            00645000
        TITLE("TOTAL ERRORS BY GROUP FOR ",A6,A6),                      00646000
        BITS(X16,"UNIT",X10,"D22",X11,"D",I2,X11,"D20",X11,"D19",       00647000
                        X11,"D18",X11,"D17",X11,"D16"),                 00648000
        RW(X8,8(X7,"W",X5,"R"),////),                                   00649000
        DETAIL(X4,A3,A5,7(A4,X2,A4,X4),A4,X2,A4);                       00650000
%                                                                       00650100
    STREAM PROCEDURE GETLUNTITLES(J,A,LUNMIN,LUNMAX,UTITLE1,UTITLE2);   00651000
```

```
        VALUE J;                                                        00651100
        BEGIN                                                           00651200
          SI:=A; J(SI:=SI+16);                                          00651300
          DI:=LUNMIN; DI:=DI+7; DS:=CHR;                                00651400
          DI:=LUNMAX; DI:=DI+7; DS:=CHR;                                00651500
          DI:=UTITLE1; DI:=DI+2; SI:=SI+1; DS:=6 CHR;                   00651600
          DI:=UTITLE2; DI:=DI+2; DS:=6 CHR;                             00651700
        END GETLUNTITLES;                                               00651800
%                                                                       00651900
        GETLUNTITLES(J,UTITLES,LUNMIN,LUNMAX,UTITLE1,UTITLE2);          00652000
LOOP:                 % FOR DRUM ERRORS                                 00653000
        IF LUNMIN > (LUN:= SNOOP(SUM(7)),[1:5]) THEN                    00653100
        BEGIN SPACE(SUM,1);                                             00653200
              GO LOOP;                                                  00653300
        END;                                                            00653400
        IF LUNMAX < LUN THEN GO EXIT;                                   00654000
        LASTL:= LUN;                                                    00655000
        LASTK:= KEY:= SNOOP(SUM(0)),[3:6];                              00656000
        WRITE(LPA[PAGE]);                                               00657000
        PRINTMFID;                                                      00657010
        WRITE(L[*],TITLE,UTITLE1,UTITLE2);                              00658000
        WRITE(LPA,15,L[*]);                                             00659000
        UNDERLINE(L);                                                   00660000
        WRITE(LPA,15,L[*]); WRITE(LPA,SPACES,4);                        00661000
        WRITE(LPA,BITS,IF J=0 THEN 36 ELSE 21);                         00662000
        WRITE(L[*],SWFORM[J]); WRITE(LPA,15,L[*]);                      00663000
        UNDERLINE(L);                                                   00664000
        WRITE(LPA[DBL],15,L[*]);                                        00665000
        WRITE(LPA,RW);                                                  00666000
COMPARE:                                                                00667000
        IF LASTL = LUN THEN                                             00667200
        BEGIN                                                           00667300
          IF LASTK = KEY THEN                                           00667400
          BEGIN                                                         00667500
ADDUP:      READ(SUM,10,A[*]); COUNT:=COUNT+1;                          00667600
            FOR I:=0 STEP 1 UNTIL 6 DO                                  00667700
            IF TESTBIT(A,I) THEN                                        00667800
            TOTERR[N:=I*2+(IO:=A[3],[24:1])]:= TOTERR[N]+1;             00667900
            UERR:= UERR+ (IF BL(IO) THEN 1 ELSE 32768);                 00668000
            LUN:= SNOOP(SUM(7)),[1:5];                                  00668100
            KEY:= SNOOP(SUM(0)),[3:6];                                  00668200
          END ELSE                                                      00668300
          BEGIN                                                         00668400
WRITEIT:    WRITE(LPA[DBL],DETAIL,TINO[LASTL],                          00668500
                          IF LASTK=2 THEN "-CONF" ELSE "      ",        00668600
                          DECL(UERR,[FF]),DECL(UERR,[CF]),              00668700
                          FOR N:=0 STEP 1 UNTIL 13 DO DECL(TOTERR[N])); 00668800
            IF LUN > LUNMAX THEN GO EXIT;                               00668900
            IF LASTL≠LUN THEN WRITE(LPA[DBL]);                          00669000
            ZERO(14,TOTERR); UERR:=0;                                   00669100
            LASTK:= KEY;                                                00669200
            LASTL:= LUN;                                                00669300
            GO ADDUP;                                                   00669400
          END;                                                          00669500
        END ELSE GO WRITEIT;                                            00669600
        GO COMPARE;                                                     00669700
EXIT:                                                                   00670000
  END GROUPTOTALS;                                                      00671000
%                                                                       00677100
PROCEDURE REPORTAPETEST;                                                00677200
```

```
BEGIN                                                                   00677300
  INTEGER R;                                                            00677400
  REAL D,P,Q,S,T,                                                       00677500
       RCNT,CHRCNT,WRDCNT,WRD,MINWRD,PTR,NEUM,                          00677600
       WC,SZ,BSIZE,CIOD,CIODCF,IOD,RESULT,                              00677700
       PATTERN,PATTERN1,PATTERN2;% DONT INSERT HERE %                   00677800
  ARRAY BMESS[0:6], BUFF[0:33];                                         00677900
  BOOLEAN MOD3IOS,BCL,NEXTOG,LASTOG,MORETOG,DUPTOG;                     00678000
  DEFINE                                                                00678100
       LAST  =BUFF[27]#,                                                00678200
       NEXT  =BUFF[21]#,                                                00678300
       PATT  =BUFF[15]#,                                                00678400
       LOCN  =BUFF[8]#,                                                 00678500
       FSTWRD=BUFF[8]#,                                                 00678600
       XLATE =BUFF[0]#,                                                 00678700
       MIN(MIN1,MIN2)= (IF MIN1<MIN2 THEN MIN1 ELSE MIN2)#;             00678900
%                                                                       00678950
  LABEL AGAIN,DUMP,MOV,EDIT,EOL,ENDTEST,EXIT;                           00679000
  FORMAT                                                                00679100
    TITLE("REAL TIME TAPE TESTS FOR UNRECOVERED WRITE FAILURES",/,      00679200
          "----- ---- ---- ----- --- ---------- ----- --------",///),   00679300
    DETAIL(A5,X3,A3,X6,"MIX=",I3,                                       00679400
           X3,"CHANNEL=",I2,X3,"PHYSICAL REEL=  ",I5),                  00679500
    OBJNAME(X88,"OBJ JOB NAME  =  ",A1,A6,"/",A1,A6),                   00679600
    FILNAME(X88,"FILE NAME     =  ",A1,A6,"/",A1,A6),                   00679700
    ORIG(/,X31,"ORIGINAL I/O DESC.    =  ",O,X1,O,                      00679800
         /,X31,"ORIGINAL RESULT DESC. =  ",O,X1,O,/),                   00679900
    RDC(X88,"TAPE NAME     =  ",A1,A6,"/",A1,A6,/,                      00680000
        X88,"REEL          =  ",I5,/,                                   00680100
        X88,"CREATION DATE =  ",A5,X2,O,/,                              00680200
        X88,"CYCLE         =  ",I5),                                    00680300
    MODE(/,X31,"(",A6,",", I4," WORDS ON BLOCK # ",I5,")"),             00680400
    TESTNO(////,"ANALYSIS FOR TEST NO. ",I2,                            00680500
           X2,"(",A6," WRITE,",A6,A4," PATTERN)",                       00680600
            /,"-------- --- ---- --- --"),                              00680700
    DESC(///,"TEST PATTERN ",A5,X9,"CHANNEL=",I2,                       00680800
          //,"I/O DESC.    =  ",O,X1,O,                                00680900
           /,"RESULT DESC. =  ",O,X1,O,/),                              00681000
    WORD(//,I4," WORDS WERE(",A3,")-",/),                               00681100
    EQLS(X8,6(2(8("=")," ")," ")),                                      00681200
    UNK(/,X27,"***** WORDS  ",I4,"  THRU  ",I4," NOT LOGGED, ",         00681300
        "THEREFORE CONTENTS UNKNOWN *****"),                            00681400
    SIZERROR(/,X16,"SIZE ERROR( ",I5," WORDS ",A*,")"),                 00681500
    NOERRORS(/,X16,"NO DATA ERRORS"),                                   00681600
    TALLY(//,I4," CHRS (IN ",I4," WDS) DIFFER FROM TEST RECORD ",       00681700
          "(FIRST IN WRD ",I4,")");                                     00681800
%                                                                       00681810
  STREAM PROCEDURE FILLPATTERN(PATTERN,K,M,N,BUFFER);                   00681900
    VALUE K,M,N;                                                        00682000
    BEGIN                                                               00682100
      DI:=BUFFER; SI:=PATTERN; SI:=SI+K;                                00682200
      M(DS:=N CHR; SI:=SI-N);                                           00682300
      SI:=BUFFER; DS:=5 WDS;                                            00682400
    END FILLPATTERN;                                                    00682500
  BOOLEAN STREAM PROCEDURE COMPARE(NEXT,LAST);                          00682600
    BEGIN                                                               00682700
      SI:=NEXT; DI:=LAST; TALLY:=1;                                     00682800
      48(IF SC≠DC THEN BEGIN TALLY:=0; JUMP OUT; END);                  00682900
      COMPARE:=TALLY;                                                   00683000
    END COMPARE;                                                        00683100
```

```
STREAM PROCEDURE BUILDPATTERN(MOD3IOS,PATTERN);                          00683200
  VALUE MOD3IOS;                                                         00683300
  BEGIN                                                                  00683400
    DI:=PATTERN;                                                         00683500
    DS:=14 LIT"01248+x←]=%X≤←";                                          00683600
    MOD3IOS(DI:=DI-7; DS:=LIT""""; DI:=DI+6);                            00683700
    DS:=LIT""""; DS:=3 LIT"]$(";                                         00683800
  END BUILDPATTERN;                                                      00683900
STREAM PROCEDURE MOVEX(BCL,HERE,THERE,XBUFF);                            00684000
  VALUE BCL;                                                             00684100
  BEGIN LABEL XIT; LOCAL SVDI,SVSI,SV;                                   00684200
    SI:=HERE; DI:=THERE;                                                 00684300
    BCL(8(SVDI:=DI; DI:=LOC SV; DI:=DI+7; DS:=CHR;                       00684400
          SVSI:=SI; SI:=XBUFF; SI:=SI+SV;                                00684500
          DI:=SVDI; DS:=CHR; SI:=SVSI));                                 00684600
        JUMP OUT TO XIT);                                                00684700
    DS:=WDS;                                                             00684800
    XIT:                                                                 00684900
  END MOVEX;                                                             00685000
STREAM PROCEDURE EDITR(N,LOCN,BUFF,L);                                   00685100
  VALUE N,LOCN;                                                          00685200
  BEGIN                                                                  00685300
    DI:=L; SI:=LOC LOCN;                                                 00685400
    DS:=4 DEC; DI:=DI+2; SI:=BUFF;                                       00685500
    N(DI:=DI+2; 8(OCTADE); DI:=DI+1; 8(OCTADE)),                         00685600
    DI:=L; DS:=3 FILL;                                                   00685700
  END EDITR;                                                             00685800
STREAM PROCEDURE GETVALUES(BUFF,CHRCNT,WRDCNT,FSTWRD);                   00685900
  BEGIN                                                                  00686000
    SI:=BUFF; SI:=SI+3;                                                  00686100
    DI:=CHRCNT; DI:=DI+5; DS:=3 CHR;                                     00686200
    DI:=WRDCNT; DI:=DI+6; DS:=2 CHR;                                     00686300
    DI:=FSTWRD; 7(DI:=DI+6; DS:=2 CHR);                                  00686400
  END GETVALUES;                                                         00686500
PROCEDURE BREAKDOWNRESULT(N,NEUM,RESULT);                                00686600
  VALUE N,NEUM; REAL N,NEUM,RESULT;                                      00686700
  BEGIN REAL Q;                                                          00686800
    IF (Q:=(ANALYZEDESC(3,NEUM,RESULT,JUNKBUFFER)-1)×5) GEQ 0 THEN       00686900
    BEGIN                                                                00687000
      BLANK(L);                                                          00687100
      FOR J:=0 STEP 5 UNTIL Q DO                                         00687200
      BEGIN                                                              00687300
        MOVE(5,JUNKBUFFER[J],L[N]);                                      00687400
        WRITE(LPA,15,L[*]);                                              00687500
    END;END;                                                             00687600
    BLANK(L);                                                            00687700
  END BREAKDOWNRESULT;                                                   00687800
%                                                                        00687900
REWIND(TST);                                                             00688000
READ(TST[NO],5,B[*]);                                                    00688100
BUILDPATTERN(MOD3IOS:=BL(B[2],[2:1]),PATTERN);                           00688300
FILL BUFF[*] WITH                                                        00688400
     OCT1201020304050607,                                                00688500
     OCT1011131400151617,                                                00688600
     OCT7261626364656667,                                                00688700
     OCT7071737460757677,                                                00688800
     OCT5241424344454647,                                                00688900
     OCT5051535440555657,                                                00689000
     OCT2021222324252627,                                                00689100
     OCT3031333432353637;                                                00689200
```

```
        FILL BMESS[*] WITH                                                      00689300
             " P-BIT",                                                          00689400
             " 1-BIT",                                                          00689500
             " 2-BIT",                                                          00689600
             " 4-BIT",                                                          00689700
             " 8-BIT",                                                          00689800
             " A-BIT",                                                          00689900
             " B-BIT";                                                          00690000
    %                                                                           00690100
    AGAIN:                                                                      00690300
          READ(TST,210,B[*]);                                                   00690400
          OIOD:= B[3];                                                          00690500
          RESULT:= B[4];                                                        00690600
          OIODCF:= OIOD.[CF];                                                   00690700
          WC:=OIOD.[8:10];                                                      00690800
          WRITE(LPA[PAGE]); PRINTMFID; WRITE(LPA,TITLE);                        00690900
          WRITE(LPA[DBL],FDATE,T:=B[206],DAY(T),ACTDATE);                       00691000
          WRITE(LPA,DETAIL                                                      00691100
                   ,ATIME(B[1].[3:24])                                          00691200
                   ,NEUM:=TINU[UNIT[OIOD.[3:5]×2]]                              00691300
                   ,MIX:=B[0].[20:5]                                            00691400
                   ,B[0].[18:2]+1                                               00691500
                   ,B[7].[30:18]                                                00691600
                  );                                                            00691700
          IF MIX≠0 THEN                                                         00691800
             WRITE(LPA,OBJNAME,B[201].[6:6],B[201],B[202].[6:6],B[202]);        00691900
          IF B[0].[9:9]≠0 THEN                                                  00692000
             WRITE(LPA,FILNAME,B[204].[6:6],B[204],B[205].[6:6],B[205]);        00692100
          WRITE(LPA,ORIG                                                        00692200
                   ,OCTCONV(0,OIOD),OCTCONV(4,OIOD)                             00692300
                   ,OCTCONV(0,RESULT),OCTCONV(4,RESULT)                         00692400
                  );                                                            00692500
          BREAKDOWNRESULT(7,NEUM,RESULT);                                       00692600
          WRITE(LPA,MODE                                                        00692700
                   ,IF BL(OIOD.[21:1]) THEN "BINARY" ELSE "ALPHA "              00692800
                   ,WC                                                          00692900
                   ,B[1].[27:21]                                                00693000
                  );                                                            00693100
          WRITE(LPA,RDC                                                         00693200
                   ,B[5].[6:6],B[5],B[6].[6:6],B[6]                             00693300
                   ,B[2].[14:10]                                                00693400
                   ,IF (D:=DECM(B[2].[24:17]))=0 THEN D:=T ELSE D               00693500
                   ,DAY(-D)                                                     00693600
                   ,B[2].[41:7]                                                 00693700
                  );                                                            00693800
          OIOD:= OIOD & 1[18:42:6];                                             00693900
          FOR I:=0 STEP 1 UNTIL 15 DO                                           00694000
          BEGIN                                                                 00694100
            WRITE(LPA,TESTNO                                                    00694200
                     ,I+1                                                       00694300
                     ,IF (BCL:=I>6 AND I<14) THEN " ALPHA" ELSE "BINARY"        00694400
                     ,IF I=14 THEN "  SKEW" ELSE IF I=15 THEN "  FLUX" ELSE   00694500
                      IF I=7 THEN IF MOD3IOS THEN BMESS[0] ELSE BMESS[6]        00694600
                      ELSE BMESS[I MOD 7]                                       00694700
                     ,IF I>13 THEN "    " ELSE IF BCL THEN "-OFF" ELSE "-ON "  00694800
                     );                                                         00694900
            WRITE(LPA,DESC,"WRITE"                                              00695000
                     ,(RESULT:=B[N:=I×12+8]).[19:2] +1                          00695100
                     ,OCTCONV(0,IOD:=OIOD&REAL(NOT BCL)[21:47:1])               00695200
                     ,OCTCONV(4,IOD)                                            00695300
```

```
            ,OCTCONV(0,RESULT),OCTCONV(4,RESULT)                          00695400
           );                                                             00695500
BREAKDOWNRESULT(2,NEUM,RESULT);                                           00695600
IF (BSIZE:=ABS(OIODCF-RESULT.[CF]))<WC                                    00695700
THEN WRITE(LPA,SIZERROR,BSIZE,5,"WRITE");                                 00695800
WRITE(LPA,WORD,WC,IF BCL THEN "BCL" ELSE "INT");                          00695900
FILLPATTERN(PATTERN                                                       00696000
            ,I+REAL(I=15),4*4×REAL(I<14),1+REAL(I>13)                     00696100
            ,PATT);                                                       00696200
IF WC>6 THEN                                                              00696300
BEGIN EDITR(6,1,PATT,L);                                                  00696400
      WRITE(LPA,15,L[*]);                                                 00696500
END;                                                                      00696600
IF WC>12 THEN WRITE(LPA,EQLS);                                            00696700
IF (R:=WC MOD 6)=0 THEN R:=6 ELSE BLANK(L);                               00696800
EDITR(R,WC+1-R,PATT,L);                                                   00696900
WRITE(LPA[DBL],15,L[*]);                                                  00697000
WRITE(LPA,DESC,"READ "                                                    00697100
            ,(RESULT:=B[N+1]).[19:2] +1                                   00697200
            ,OCTCONV(0,IOD:=IOD&1[24:47:1])                               00697300
            ,OCTCONV(4,IOD)                                               00697400
            ,OCTCONV(0,RESULT), OCTCONV(4,RESULT)                         00697500
           );                                                             00697600
BREAKDOWNRESULT(2,NEUM,RESULT);                                           00697700
GETVALUES(B[N+2],CHRCNT,WRDCNT,LOCN);                                     00697800
SZ:= IF (BSIZE:=ABS(OIODCF-RESULT.[CF]))>WC THEN WC ELSE                  00697900
     IF BL(RESULT.[21:1]) THEN BSIZE ELSE                                 00698000
     BSIZE - REAL(RESULT.[15:3]=0);                                       00698100
IF SZ≠WC THEN WRITE(LPA,SIZERROR,SZ,4,"READ");                            00698200
IF WRDCNT=0 THEN WRITE(LPA,NOERRORS) ELSE                                 00698300
BEGIN                                                                     00698400
  WRITE(LPA,WORD,SZ,IF BCL THEN "BCL" ELSE "INT");                        00698500
  D:= SZ DIV 6 - REAL((R:=SZ MOD 6)=0);                                   00698600
  IF R=0 THEN R:=6;                                                       00698700
  WRD:=0; PTR:=8;                                                         00698800
  MINWRD:=MIN(WRDCNT,7)+7;                                                00698900
  MORETOG:= WRDCNT>7;                                                     00699000
  FOR P:=0 STEP 1 UNTIL D DO                                              00699100
  BEGIN                                                                   00699200
    MOVE(6,PATT,NEXT);                                                    00699300
    NEXTOG:=FALSE;                                                        00699400
    FOR T:=PTR STEP 1 UNTIL MINWRD DO                                     00699500
    BEGIN                                                                 00699600
      IF (S:=BUFF[T]) > WRD+5 THEN T:=MINWRD ELSE                         00699700
      BEGIN                                                               00699800
        MOVEX(BCL,B[N+T-3],BUFF[21+S-WRD],XLATE);                         00699900
        IF (PTR:=PTR+1)>14 THEN IF MORETOG THEN                           00700000
        BEGIN                                                             00700100
          BLANK(L);                                                       00700200
          EDITR(S+1-WRD,WRD+1,NEXT,L);                                    00700300
          WRITE(LPA,15,L[*]);                                             00700400
          WRITE(LPA,UNK,S+2,SZ);                                          00700500
          GO ENDTEST;                                                     00700600
        END;                                                              00700700
        NEXTOG:=TRUE;                                                     00700800
      END;                                                                00700900
    END;                                                                  00701000
    IF P=0 OR P=D THEN BEGIN BLANK(L); GO MOV; END ELSE                   00701100
    IF NEXTOG THEN                                                        00701200
    IF LASTOG THEN                                                        00701300
```

```
            IF COMPARE(NEXT,LAST) THEN                                  00701400
          DUPE:IF DUPTOG THEN GO EOL ELSE BEGIN WRITE(L[*],EQLS);       00701500
                                              DUPTOG:=TRUE;             00701600
                                          END                           00701700
            ELSE BEGIN                                                  00701800
          MOV:      MOVE(6,NEXT,LAST);                                  00701900
          EDIT:     LASTOG:=NEXTOG;                                     00702000
                    EDITR(IF P#0 THEN 6 ELSE R,WRD+1,NEXT,L);           00702100
                    DUPTOG:=FALSE;                                      00702200
                END                                                     00702300
            ELSE GO MOV                                                 00702400
            ELSE IF LASTOG THEN GO EDIT ELSE GO DUPE;                   00702500
            WRITE(LPA,15,L[*]);                                         00702600
          EOL: WRD:=WRD+6;                                              00702700
          END;                                                          00702800
          ENDTEST:                                                      00702900
            WRITE(LPA,TALLY,CHRCNT,WRDCNT,FSTWRD+1);                    00703000
          END;                                                          00703100
        END;                                                            00703200
        IF (RCNT:=RCNT+1)#TAPETESTS THEN GO AGAIN;                      00703300
EXIT:                                                                   00703500
        REWIND(TST);                                                    00703700
      END REPORTAPETEST;                                                00703800
%                                                                       00742000
PROCEDURE HI(A); ARRAY A[0];                                            00743000
  A[0]:=A[1]:=A[7]:=MARKER;                                             00744000
%                                                                       00745000
BOOLEAN PROCEDURE CMP(A,B); ARRAY A,B[0];                               00746000
CMP    :=IF (T1:=A[7].[7:26]) < (T2:=B[7].[7:26]) THEN TRUE ELSE        00749000
          IF T1 > T2 THEN FALSE ELSE                                    00750000
          IF (T1:=A[0].[3:6]) < (T2:=B[0].[3:6]) THEN TRUE ELSE         00750100
          IF T1 > T2 THEN FALSE ELSE                                    00750200
          IF (T1:=A[1].[27:21]) < (T2:=B[1].[27:21]) THEN TRUE ELSE     00750300
          IF T1 > T2 THEN FALSE ELSE                                    00750400
          IF (T1:=A[7].[33:6]) < (T2:=B[7].[33:6]) THEN TRUE ELSE       00750500
          IF T1 > T2 THEN FALSE ELSE                                    00750600
          IF (T1:=A[7].[39:6]) < (T2:=B[7].[39:6]) THEN TRUE ELSE       00751000
          IF T1 > T2 THEN FALSE ELSE                                    00752000
          IF (T1:=A[7].[45:3]) < (T2:=B[7].[45:3]) THEN TRUE ELSE       00753000
          IF T1 > T2 THEN FALSE ELSE TRUE;                              00754000
%                                                                       00754100
BOOLEAN PROCEDURE CMPLUN(A,B); ARRAY A,B[0];                            00755000
CMPLUN:=IF (T1:=A[7].[1:5]) < (T2:=B[7].[1:5]) THEN TRUE ELSE           00755100
          IF T1 > T2 THEN FALSE ELSE                                    00755200
          IF (T1:=A[0].[3:6]) < (T2:=B[0].[3:6]) THEN TRUE ELSE         00755300
          IF T1 > T2 THEN FALSE ELSE                                    00755400
          IF (T1:=A[1].[27:21]) < (T2:=B[1].[27:21]) THEN TRUE ELSE     00755500
          IF T1 > T2 THEN FALSE ELSE                                    00755600
          IF (T1:=A[1].[3:24]) < (T2:=B[1].[3:24]) THEN TRUE ELSE       00755700
          IF T1 > T2 THEN FALSE ELSE TRUE;                              00755800
%                                                                       00755900
BOOLEAN PROCEDURE INLUNSORT(A); ARRAY A[0];                             00756000
  BEGIN                                                                 00757000
    READ(SWIN[J],10,A[*]);                                              00758000
    IF A[7]=MARKER THEN                                                 00759000
    BEGIN REWIND(SWIN[J]);                                              00760000
          INLUNSORT:= TRUE;                                             00761000
  END; END INLUNSORT;                                                   00762000
%                                                                       00762050
BOOLEAN PROCEDURE INSORT(A); ARRAY A[0];                                00762100
```

```
      BEGIN                                                                    00762200
        IF SNOOP(SUM(7)),[1:5] > LUNMAX THEN INSORT:= TRUE                     00762300
        ELSE READ(SUM,10,A[*]);                                                00762400
      END INSORT;                                                              00762500
    %                                                                          00762600
    PROCEDURE OUTSORT(B,A); VALUE B; BOOLEAN B; ARRAY A[0];                    00763000
      BEGIN                                                                    00764000
        IF B THEN                                                              00765000
        BEGIN JUNKBUFFER[0]:=JUNKBUFFER[7]:= MARKER;                           00766000
              WRITE(SWOUT[J],10,JUNKBUFFER[*]);                                00767000
              REWIND(SWOUT[J]);                                                00768000
        END ELSE WRITE(SWOUT[J],10,A[*]);                                      00769000
      END OUTSORT;                                                             00770000
    %                                                                          00770100
    PROCEDURE REPORTPRN;                                                       00771000
      BEGIN                                                                    00772000
        ARRAY TOTERR[0:15];                                                    00773000
        REAL IO,LASTPRN,PRN,LASTKEY,KEY,UERR;                                  00774000
        LABEL AGAIN,COMPARE,ADDUP,WRITEIT,EXIT;                                00775000
        BOOLEAN TOG;                                                           00776000
        FORMAT                                                                 00777000
              TITLE("TAPE ERRORS BY PHYSICAL REEL NUMBER",/,                   00778000
                    "----- ------ -- -------- ---- ------",////),              00779000
              BITS (X17,"PRN",X10,"D22",X11,"D36",X11,"D20",X11,"D19",         00780000
                                                                               00781000
                          X11,"D18",X11,"D17",X11,"D16",/,                     00782000
                    X5,"PRN",X7,"TOTALS",X5,"MEM ADDRESS",X4,"BLANK TAPE",     00783000
                    X3,"TAPE PARITY",X4,"MEM PARITY",X5,"NOT READY",           00784000
                    X3,"DESC PARITY",X7,"BUSY",/,                              00785000
                    X5,"---",X7,"------",X5,"--- --------",X4,"------ ----",   00786000
                    X3,"---- ------",X4,"--- ------",X5,"--- -----",           00787000
                    X3,"---- ------",X7,"----",//,                             00788000
                    X8,8(X7,"W",X5,"R"),////),                                 00789000
              UNITS(X5,"PRN",X5,"TOTALS",                                      00790000
                    X8,"MTA",X3,"MTB",X3,"MTC",X3,"MTD",X3,"MTE",X3,           00791000
                       "MTF",X3,"MTH",X3,"MTJ",X3,"MTK",X3,"MTL",X3,           00792000
                       "MTM",X3,"MTN",X3,"MTP",X3,"MTR",X3,"MTS",X3,"MTT",/    00793000
                    X5,"---",X5,"------",X5,16(X3,"---"),////),                00794000
              SUB(X2,I5,A5,7(A4,X2,A4,X4),A4,X2,A4),                           00795000
              DETAIL(X2,I5,A5,X2,A4,X6,16(X2,A4));                             00796000
    AGAIN:                                                                     00797000
        WRITE(LPA[PAGE]);                                                      00798000
        PRINTMFID ;                                                            00798010
        WRITE(LPA,TITLE);                                                      00799000
        IF TOG THEN WRITE(LPA,UNITS) ELSE WRITE(LPA,BITS);                     00800000
        ZERO(16,TOTERR); UERR:=0;                                              00801000
        READ(SORTED,10,A[*]);                                                  00802000
        LASTPRN:= A[7].[15:18];                                                00803000
        LASTKEY:= A[0].[3:6];                                                  00804000
    COMPARE:                                                                   00805000
        IF LASTPRN = (PRN:=A[7].[15:18]) THEN                                  00806000
        BEGIN                                                                  00807000
          IF LASTKEY= A[0].[3:6] THEN                                          00808000
          BEGIN                                                                00809000
    ADDUP:  IF TOG THEN                                                        00810000
            BEGIN                                                              00811000
              TOTERR[LUN:=A[7].[2:4]]:= TOTERR[LUN]+1;                         00812000
              UERR:=UERR+1;                                                    00813000
            END ELSE                                                           00814000
            BEGIN                                                              00815000
```

```
            FOR J:=0 STEP 1 UNTIL 6 DO                                          00816000
            IF TESTBIT(A,J) THEN                                                00817000
            TOTERR[N:=J*2+(IO:=A[3].[24:1]]]:= TOTERR[N]+1;                     00818000
            UERR:= UERR+ (IF BL(IO) THEN 1 ELSE 32768);                         00819000
          END;                                                                  00820000
          READ(SORTED,10,A[*]);                                                 00821000
        END ELSE                                                                00822000
        BEGIN                                                                   00823000
WRITEIT: IF TOG THEN                                                            00824000
          WRITE(LPA[DBL],DETAIL,LASTPRN,                                        00825000
                  IF LASTKEY=21 THEN "-CONF" ELSE "     ",                      00826000
                  DECL(UERR),                                                   00827000
                  FOR N:=0 STEP 1 UNTIL 15 DO DECL(TOTERR[N]))                  00828000
          ELSE                                                                  00829000
          WRITE(LPA[DBL],SUB,LASTPRN,                                           00830000
                  IF LASTKEY=21 THEN "-CONF" ELSE "     ",                      00831000
                  DECL(UERR.[FF]), DECL(UERR.[CF]),                             00832000
                  FOR N:=0 STEP 1 UNTIL 13 DO DECL(TOTERR[N]));                 00833000
          IF PRN > 99999 THEN GO EXIT;                                          00834000
          IF LASTPRN# PRN THEN WRITE(LPA[DBL]);                                 00835000
          ZERO(16,TOTERR); UERR:=0;                                             00836000
          LASTPRN:= PRN;                                                        00837000
          LASTKEY:= A[0].[3:6];                                                 00838000
          GO ADDUP;                                                             00839000
        END                                                                     00840000
      END ELSE GO WRITEIT;                                                      00841000
      GO COMPARE;                                                               00842000
EXIT:                                                                           00843000
      REWIND(SORTED);                                                           00844000
      IF NOT TOG THEN                                                           00845000
      BEGIN TOG:= TRUE;                                                         00846000
            GO AGAIN;                                                           00847000
      END;                                                                      00848000
    END REPORTPRN;                                                              00849000
%                                                                               00850000
PROCEDURE REPORTBADISK;                                                         00850020
BEGIN                                                                           00850040
      REAL LASTEU,LASTSU;                                                       00850060
      ARRAY BREAKDOWN[C:9];                                                     00850080
      DEFINE                                                                    00850100
      BK    =BREAKDOWN#,                                                        00850120
      EU    =BK[1]#,                                                            00850140
      SU    =BK[2]#,                                                            00850160
      SET   =BK[3]#,                                                            00850180
      ZONE  =BK[5]#,                                                            00850200
      TRACK=BK[6]#,                                                             00850220
      HEAD  =BK[8]#,                                                            00850240
      PIN   =BK[9]#;                                                            00850260
      LABEL AGAIN,LINE1,SETUP,EXIT;                                             00850280
      FORMAT                                                                    00850300
      TITLE("BADISK AREAS BY DISK ADDRESS FOR EU ",I2,/                         00850320
            "------ ----- -- ---- ------- --- --  -",////),                    00850340
      HEADING(X4,"BEGINNING",/,                                                 00850360
              X4,"ADDRESS     SEGS     PREVIOUS FILE",X6,"DISK  ZONE  ",      00850380
              "TRACK  HEAD  PIN",/,                                             00850400
              X4,"-------     ----     --------  -","---",X6,"----  ----  ",  00850420
              "-----  ----  ---",////),                                         00850440
      DETAIL(X4,A1,A6,X1,I8,X5,A1,A6,A1,A1,A6,X6,                               00850460
      I1,X5,I1,X5,A2,X4,I2,X3,I2),                                              00850480
      SUF("SU ",I1,//);                                                         00850500
```

```
%                                                                    00850520
      LASTEU:=64;                                                    00850540
AGAIN:                                                               00850560
      READ(SORTED,10,A[*]);                                          00850580
      IF A[7]#MARKER THEN BREAKDOWNADDRESS(A[2],BK)                  00850600
      ELSE GO EXIT;                                                  00850620
      IF EU=LASTEU THEN                                              00850640
      BEGIN                                                          00850660
            IF SU=LASTSU THEN                                        00850680
LINE1:      WRITE(LPA,DETAIL                                         00850700
                  ,A[2].[6:6],A[2]                                   00850720
                  ,ABS(A[4])                                         00850740
                  ,A[5].[6:6],A[5]                                   00850760
                  ,IF A[4] LSS 0 THEN "/" ELSE " "                   00850780
                  ,A[6].[6:6],A[6]                                   00850800
                  ,SET,ZONE,DECM(TRACK),HEAD,PIN                     00850820
                 )                                                   00850840
            ELSE                                                     00850860
            BEGIN                                                    00850880
                  WRITE(LPA[DBL]);                                   00850900
SETUP:            WRITE(LPA,SUF,SU);                                 00850920
                  LASTSU:=SU;                                        00850940
                  LASTEU:=EU;                                        00850960
                  GO LINE1;                                          00850980
            END                                                      00851000
      END ELSE                                                       00851020
      BEGIN                                                          00851040
            WRITE(LPA[PAGE]);                                        00851060
            PRINTMFID;                                               00851080
            WRITE(LPA,TITLE,EU);                                     00851100
            WRITE(LPA,HEADING);                                      00851120
            GO SETUP;                                                00851140
      END;                                                           00851160
      WRITE(LPA);                                                    00851180
      GO AGAIN;                                                      00851200
EXIT:                                                                00851220
      REWIND(SORTED);                                                00851240
END REPORTBADISK;                                                    00851260
%                                                                    00852000
PROCEDURE REPORTADDRESS;                                             00853000
  BEGIN                                                              00854000
    REAL I,J,NEUM,MAINT,IO,REPEATS,                                  00855000
         LASTEU,LASTSU,SUERR,LASTKEY,LASTERROR,                      00855100
    REAL LASTADR;                                                    00855150
    ARRAY BREAKDOWN[0:9];                                            00855200
    DEFINE EU    =BREAKDOWN[1]#,                                     00855300
           SU    =BREAKDOWN[2]#,                                     00855400
           SET   =BREAKDOWN[3]#,                                     00855500
           ZONE  =BREAKDOWN[5]#,                                     00855600
           TRACK=BREAKDOWN[6]#,                                      00855700
           HEAD  =BREAKDOWN[8]#,                                     00855800
           PIN   =BREAKDOWN[9]#;                                     00855900
    BOOLEAN ADDRESSERR,DUPADR;                                       00856000
    LABEL AGAIN,LINE1,PEEP,SETUP,EXIT;                               00856100
    FORMAT                                                           00857000
      TITLE("DISK ERRORS BY DISK ADDRESS FOR EU ",I2,/               00858000
            "---- ------ -- ---- ------- --- --  -",////),           00859000
      DISKADDRESSERROR("DISK ADDRESS ERRORS",/,                      00859100
                       "---- ------- ------",////),                 00859200
      HEADING(X4,"BEGINNING",/,                                      00860000
```

```
                X4,"ADDRESS   SEGS   MAINT   REPEAT",X9,"DISK  ZONE  ",    00861000
                  "TRACK  HEAD  PIN  RETRY  I/O  UNIT      ERRORS",/,    00862000
                X4,"-------  ----   -----   ------",X9,"----  ----  ",    00863000
                  "------  ----  ---  -----  ---  ----     -------",////),00864000
        DETAIL(X4,A1,A6,X4,I2,X6,A2,X4,A6,X11,A1,X5,A1,X5,                 00865000
               A2,X4,A2,X3,A2,X5,I2,A1,X3,A1,X3,A3," (",A1,")"),           00866000
        SUTOT(/,X14,"TOTAL ERRORS = ",I6,//),                              00867000
        SUF("SU ",I1,//);                                                  00868000
%                                                                          00868100
    LASTEU:=64;                                                            00869000
AGAIN:                                                                     00870000
    READ(SORTED,10,A[*]);                                                  00871000
    IF A[7]≠MARKER THEN IF ADDRESSERR THEN GO LINE1 ELSE                   00872000
    IF BL(A[7].[7:1]) THEN                                                 00872100
    BEGIN IF SUERR≠0 THEN WRITE(LPA,SUTOT,SUERR);                          00872200
          WRITE(LPA[PAGE]);                                                00872300
          PRINTMFID;                                                       00872310
          WRITE(LPA,DISKADDRESSERROR);                                     00872400
          WRITE(LPA,HEADING);                                              00872500
          ZERO(10,BREAKDOWN); SUERR:=0;                                    00872600
          ADDRESSERR:= TRUE;                                               00872700
          DUPADR:= FALSE;                                                  00872750
          GO LINE1;                                                        00872800
    END ELSE                                                               00873000
    BEGIN                                                                  00873010
         BREAKDOWNADDRESS(A[2],BREAKDOWN);                                 00873020
         IF LASTADR=A[7].[8:25] THEN DUPADR:=TRUE ELSE                     00873030
         BEGIN                                                             00873040
              DUPADR:=FALSE;                                               00873050
              LASTADR:=A[7].[8:25];                                        00873060
          END;                                                             00873070
    END ELSE                                                               00873080
    IF ADDRESSERR THEN                                                     00873100
    BEGIN WRITE(LPA,SUTOT,SUERR);                                          00873200
          GO EXIT;                                                         00873300
    END ELSE EU:=63;                                                       00873400
    IF EU=LASTEU THEN                                                      00874000
    BEGIN                                                                  00875000
      IF SU=LASTSU THEN                                                    00876000
      BEGIN                                                                00877000
LINE1:  LASTKEY:=A[7];                                                     00877100
        LASTERROR:=A[4].[25:8];                                            00877200
        IO:=A[4].[24:1];                                                   00877300
        TYPE:=A[0].[3:6];                                                  00877400
        MAINT:=A[2].[42:2];                                                00877500
PEEP:   READ(SORTED[NO],10,L[*]);                                          00877600
        IF LASTKEY=L[7] THEN                                               00877700
        IF LASTERROR=L[4].[25:8] THEN                                      00877800
        IF IO=L[4].[24:1] THEN                                             00877900
        IF TYPE=L[0].[3:6] THEN                                            00878000
        IF MAINT=L[2].[42:2] THEN                                          00878100
        BEGIN                                                              00878200
          REPEATS:= REPEATS+1;                                             00878300
          SUERR:=SUERR+1;                                                  00878400
          READ(SORTED);                                                    00878500
          GO PEEP;                                                         00878600
        END;                                                               00878700
        IF DUPADR THEN                                                     00878710
        BEGIN                                                              00878720
            A[2]:="       ";                                               00878730
```

```
            ZERO(10,BREAKDOWN);                                                 00878740
          END ELSE                                                              00878750
          IF TYPE=21 THEN A[2],[42:2]:=0;                                       00878800
          WRITE(L[*],DETAIL                                                     00878900
                    ,A[2].[6:6],A[2]                                            00879000
                    ,A[3].[2:6]                                                 00879100
                    ,IF TYPE=21 THEN IF MAINT=1 THEN "M1" ELSE                  00879200
                     IF MAINT=3 THEN "M2" ELSE "CL" ELSE "  "                   00879300
                    ,DECL(REPEATS+REAL(REPEATS≠0))                              00879400
                    ,DECL(SET),DECL(ZONE)                                       00879500
                    ,IF ADDRESSERR OR DUPADR THEN "  " ELSE DECM(TRACK)        00879510
                    ,DECL(HEAD),DECL(PIN)                                       00879520
                    ,I:=A[7].[39:6]                                             00879600
                    ,IF I=(IF TYPE=21 THEN 4 ELSE 10) THEN "*" ELSE " "         00879650
                    ,DECL(A[7].[45:3])                                          00879700
                    ,NEUM:=TINU[A[7].[1:5]]                                     00879800
                    ,IF IO=1 THEN 41 ELSE 54                                    00879900
                   );                                                           00880000
          IF (K:=(ANALYZEDESC(5,NEUM,A[4],JUNKBUFFER)-1)×5) GEQ 0 THEN          00880100
          BEGIN FOR J:=0 STEP 5 UNTIL K DO                                      00880200
                BEGIN MOVE(3,JUNKBUFFER[J],L[12]);                              00880300
                  WRITE(LPA,15,L[*]);                                           00880400
                  BLANK(L);                                                     00880500
          END;  END;                                                            00880600
          SUERR:=SUERR+1;                                                       00880800
          REPEATS:=0;                                                           00880900
        END ELSE                                                                00881000
        BEGIN                                                                   00882000
          WRITE(LPA,SUTOT,SUERR);                                               00882100
SETUP:    WRITE(LPA,SUF,SU);                                                    00882200
          SUERR:=0;                                                             00882300
          LASTSU:= SU;                                                          00882400
          LASTEU:= EU;                                                          00882500
          GO LINE1;                                                             00882600
        END                                                                     00882700
      END ELSE                                                                  00883000
      BEGIN                                                                     00884000
        IF LASTEU≠ 64 THEN WRITE(LPA,SUTOT,SUERR);                              00885000
        IF EU=63 THEN GO EXIT;                                                  00886000
        WRITE(LPA[PAGE]);                                                       00887000
        PRINTMFID ;                                                             00887010
        WRITE(LPA,TITLE,EU);                                                    00888000
        WRITE(LPA,HEADING);                                                     00889000
        GO SETUP;                                                               00890000
      END;                                                                      00891000
      WRITE(LPA);                                                               00892000
      GO AGAIN;                                                                 00893000
EXIT:                                                                           00894000
      REWIND(SORTED);                                                           00895000
    END REPORTADDRESS;                                                          00896000
%                                                                               00896100
PROCEDURE INITIALIZE;                                                           00896200
BEGIN LABEL LOOP,EOFC;                                                          00896300
      FORMAT                                                                    00896400
          FSTOP("STOP←"),                                                       00896410
          FCOMMON("WHICH COMMON...←"),                                          00896420
          FCOMMONVALUES("COMMON VALUES AVAILABLE ARE:"/                         00896430
                        "01 GENERATE ANY AND ALL REPORTS"/                      00896440
                        "11 CHRONOLOGICAL LISTING ONLY"/                        00896450
                        "21 PERIPHERIAL UNIT ERROR SUMMARY"/                    00896460
```

```
                    "41 REAL TIME TAPE WRITE FAILURES"/                    00896470
                    "81 ONLINE CONFIDENCE TEST RESULTS←") ;                00896480
%                                                                          00896500
PROCEDURE MAKEINPUTLOGMAK ;                                                00897000
BEGIN LABEL NEXT;                                                          00897040
INTEGER SER,COUNT;                                                         00897060
FORMAT                                                                     00897130
     FSEC(A1,A6,"/",A1,A6," SECURED FILE --- MAINT LOGGER←"),              00897140
     FMAIN("MAINT/",A4,"←"),                                               00897160
     FFREE("** PLEASE FREE =/MNTLOG AND RESTART ***←"),                    00897200
     FNON("** NO =/MNTLOG FILES FOUND, ",                                  00897240
          "ONLY MAINT/LOG WILL BE PROCESSED ***←"),                        00897260
     FOK(A1,A6,"/",A1,A6," FILE OK --- MAINT LOGGER←") ;                   00897280
%                                                                          00897300
STREAM PROCEDURE MAKEMFID(SER,MFID); VALUE SER;                            00897320
BEGIN                                                                      00897340
     DI:=MFID; DI:=DI+5;                                                   00897360
     SI:=LOC SER; DS:=3 DEC;                                               00897380
END MAKEMFID ;                                                             00897400
%                                                                          00897420
IF COMMON=0 THEN                                                           00897440
BEGIN                                                                      00897460
     MFID:=0&TIME(5)[6:12:24];                                             00897480
END ELSE                                                                   00897500
MFID:=0&DECM(COMMON)[6:24:24];                                             00897510
FID := LOGFID ;                                                            00897520
FOR SER := 1 STEP 1 UNTIL 999 DO                                           00897540
BEGIN                                                                      00897560
     MAKEMFID(SER,MFID) ;                                                  00897580
     FILL LOG WITH MFID,FID ;                                              00897600
     SEARCH(LOG,A[*]) ;                                                    00897620
     IF A[0] = -1 THEN GO TO NEXT ;                                        00897640
     IF A[0] = 0 THEN                                                      00897660
     BEGIN                                                                 00897680
          WRITE(SPO,FSEC,LMFID);                                           00897700
          WRITE(SPO,FFREE) ; GO TO FINIS ;                                 00897720
     END;                                                                  00897740
     WRITE(A[*],FOK,LMFID);                                                00897750
     WRITE(SPO,10,A[*]);                                                   00897760
     WRITE(FINPUT,10,A[*]);                                                00897780
     COUNT := COUNT + 1 ;                                                  00897800
NEXT:                                                                      00897820
   END;                                                                    00897840
   IF COUNT = 0 THEN                                                       00897860
   BEGIN                                                                   00897880
     WRITE(SPO,FNON);                                                      00897900
   END;                                                                    00897920
   IF COMMON=0 THEN                                                        00897940
   WRITE(FINPUT,FMAIN,FID) ;                                               00897960
   COMMON := 22 ;                                                          00898000
   USEFINPUT := TRUE ;                                                     00898020
   END MAKEINPUTLOGMAK;                                                    00898060
%                                                                          00899000
  TIME0:=TIME(0);                                                          00899100
  MARKER:=REAL(NOT FALSE);                                                 00899200
  LOG.MYUSE:=INPUT;                                                        00899400
  IF TIME(-1)="SITE    " AND                                               00899500
     (COMMON=0 OR COMMON>100) THEN                                         00899600
  MAKEINPUTLOGMAK;                                                         00899700
%                                                                          00899800
```

```
        IF COMMON=0 THEN                                                    00899900
        BEGIN FILL SPO WITH *,*,*,*,*,*,19;                                 00900000
              WRITE(SPO[STOP],FCOMMON);                                     00901000
              READ(SPO,//,COMMON);                                          00902000
              IF COMMON = 0 THEN                                            00902100
              BEGIN                                                         00902110
                   WRITE(SPO,FCOMMONVALUES) ;                               00902120
                   WRITE(SPO[STOP],FCOMMON) ;                               00902130
                   READ(SPO,//,COMMON) ;                                    00902140
              END ;                                                         00902150
        END;                                                                00903000
        IF COMMON > 9 THEN                                                  00904000
        BEGIN COM:=COMMON DIV 10;                                           00905000
              COMMON:=COMMON MOD 10;                                        00906000
              LISTOG:= BL(COM).[47:1];                                      00906100
              SUMTOG:= BL(COM).[46:1];                                      00906200
              TESTOG:= BL(COM).[45:1];                                      00906300
              ONLTOG:= BL(COM).[44:1];                                      00906400
        END ELSE                                                            00907000
        LISTOG:=SUMTOG:=TESTOG:=ONLTOG:=TRUE;                               00908000
        IF COMMON=2 THEN                                                    00909000
        BEGIN                                                               00909100
              IF USEFINPUT THEN GO EOFC;                                    00909150
  LOOP: READ(WHATLOG,10,A[*])[EOFC];                                        00909200
              WRITE(FINPUT,10,A[*]);                                        00909300
              GO LOOP;                                                      00909400
  EOFC: CLOSE(WHATLOG);                                                     00909500
              WRITE(FINPUT,FSTOP);                                          00909600
              REWIND(FINPUT);                                               00909700
        END;                                                                00909800
        FILL TINU[*] WITH                                                   00909900
              "MTA","MTB","MTC","MTD","MTE","MTF","MTH","MTJ",              00910000
              "MTK","MTL","MTM","MTN","MTP","MTR","MTS","MTT",              00911000
              "DRA","DRB","DKA","DKB","LPA","LPB","CPA","CRA",              00912000
              "CRB","SPO","PPA","PRA","PRB","PPB","DCA","UNK";              00913000
        LOCK(TINU[*]);                                                      00913100
        FILL UNK[*] WITH                                                    00913200
              "UNK","CR ","LP ","MT ","DR ","DK ",                          00913300
              "SPO","CP ","UNK","PP ","PR ","DCA";                          00913400
        LOCK(UNK[*]);                                                       00913500
        FILL UNIT[*] WITH                                                   00914000
  %           WRITE - READ                                                  00915000
              31,     31,   %  0  ---, ---                                  00916000
               0,      C,   %  1  MTA                                       00917000
              31,     31,   %  2  ---, ---                                  00918000
               1,      1,   %  3  MTB                                       00919000
              16,     16,   %  4  DRA                                       00920000
               2,      2,   %  5  MTC                                       00921000
              18,     18,   %  6  DKA                                       00922000
               3,      3,   %  7  MTD                                       00923000
              17,     17,   %  8  DRB                                       00924000
               4,      4,   %  9  MTE                                       00925000
              22,     23,   %  10 CPA, CRA                                  00926000
               5,      5,   %  11 MTF                                       00927000
              19,     19,   %  12 DKB                                       00928000
               6,      6,   %  13 MTH                                       00929000
              31,     24,   %  14 ---, CRB                                  00930000
               7,      7,   %  15 MTJ                                       00931000
              30,     30,   %  16 DCA                                       00932000
               8,      8,   %  17 MTK                                       00933000
```

```
            26,     27,    % 18  PPA, PRA                                  00934000
             9,      9,    % 19  MTL                                       00935000
            29,     28,    % 20  PPB, PRB                                  00936000
            10,     10,    % 21  MTM                                       00937000
            20,     31,    % 22  LPA, ---                                  00938000
            11,     11,    % 23  MTN                                       00939000
            31,     31,    % 24  ---, ---                                  00940000
            12,     12,    % 25  MTP                                       00941000
            21,     31,    % 26  LPB, ---                                  00942000
            13,     13,    % 27  MTR                                       00943000
            31,     31,    % 28  ---, ---                                  00944000
            14,     14,    % 29  MTS                                       00945000
            25,     25,    % 30  SPO                                       00946000
            15,     15;    % 31  MTT                                       00947000
    LOCK(UNIT[*]);                                                         00947100
    FILL MESS[*] WITH                                                      00948000
        "0*NONAME", "0*HALTED",    % 0                                      00949000
        "0*NONAME", "0*ENDLOG",    % 2                                      00950000
        "RECOVERE", "D       ",    % 4                                    00951000
        "UNRECOVE", "RED     ",    % 6                                    00952000
        "OPPOSITE", " MODE   ",    % 8                                    00953000
        "BLANK TA", "PE ABORT",    % 10                                   00954000
        "EOT ON W", "RITE    ",    % 12                                   00955000
        "0 FLAG  ", "0  BIT  ",    % 14                                   00955100
        "0 INVALD", "0  INDEX",    % 16                                   00955200
        "0 INTEGR", "0  OVFLW",    % 18                                   00955300
        "0 EXPON ", "0  OVFLW",    % 20                                   00955400
        "0 DIV BY", "0  ZERO ",    % 22                                   00955500
        "0 END OF", "0  FILE ";    % 24                                   00955600
    LOCK(MESS[*]);                                                         00955900
    FILL UTITLES[*] WITH                                                   00956000
        OCT0017104421276063, "APE     ",  % 0 MAG TAPE                    00957000
        OCT2223042431624260, "        ",  % 1 DISK                        00958000
        OCT2021042451644460, "        ",  % 2 DRUM                        00959000
        OCT2425074751314563, "ER      ",  % 3 PRINTER                     00960000
        OCT2626122321512460, "PUNCH   ",  % 4 CARD PUNCH                  00961000
        OCT2730132321512460, "READER  ",  % 5 CARD READER                 00962000
        OCT3131036247406060, "        ",  % 6 SPO                         00963000
        OCT3235124721472551, " TAPE   ",  % 7 PAPER TAPE                  00964000
        OCT3636112421632160, "COMM    ";  % 8 DATA COMM                   00965000
    LOCK(UTITLES[*]);                                                      00965100
    FILL DAYS[*] WITH                                                      00966000
        "   SUN", "   MON", "  TUES", "WEDNES",                           00967000
        " THURS", "   FRI", " SATUR";                                      00968000
    LOCK(DAYS[*]);                                                         00968100
%                                                                          00970000
    FLAG:=FLAGBIT;                                                         00973100
    INDEX:=INVALDINDEX;                                                    00973200
    INTOVR:=INTEGROVFLW;                                                   00973300
    EXPOVR:=EXPONOVFLW;                                                    00973400
    DIVZERO:=DIVBYZERO;                                                    00973500
END INITIALIZE;                                                            00973600
%                                                                          00973700
PROCEDURE PASS1;                                                           00974000
BEGIN                                                                      00974100
      REAL MAXTYPE,LOGVERSION;                                             00974200
      FORMAT                                                               00974300
        WFILE("WHICH LOG FILE...←"),                                       00974400
        NFILE("#NO FIL ON DISK ",A1,A6,"/",A1,A6,"←"),                     00974500
        FTIME(X8,"TIME IS   ",A5),                                         00974600
```

```
      WM(X8,"MCPBASE IS ",I8),                                                  00974700
      NEU(X8,"NO. OF EUS ON LINE ARE",I3),                                      00974800
      MODEL(X8,"EU",I3,"=-DISK MODEL ",A2," (",I2," MS)");                      00974900
1   %                                                                           00975000
2   LABEL                                                                       00976000
3     PERIP,OBJOB,FILEN,ECJOB,HALOD,WMCP,TMDAT,XDISK,FECOM,                     00977000
4     TAPETST,                                                                  00977100
5     ONLTEST,                                                                  00977200
6     FINDIT,REED,CROSSREF,EXIT;                                                00978000
7   %                                                                           00978900
8   SWITCH SWTYPE:=                                                             00979000
9     PERIP,PERIP,PERIP,PERIP,PERIP,PERIP,PERIP,REED,                           00980000
10    PERIP,PERIP,PERIP,                                                        00981000
11    OBJOB,FILEN,ECJOB,HALOD,WMCP,TMDAT,XDISK,FECOM,                           00982000
12    TAPETST,                                                                  00982100
13    ONLTEST,                                                                  00982200
14    REED;                                                                     00982900
15  %                                                                           00982950
16  STREAM PROCEDURE MOVEMESS(PTR,L);                                           00983000
17    BEGIN LABEL LL;                                                           00984000
18      SI:=PTR; DI:=L;                                                         00985000
19      2(52(IF SC≠ARROW THEN DS:=CHR ELSE JUMP OUT 2 TO LL));                  00986000
20      LL: DS:=LIT",";                                                         00987000
21    END MOVEMESS;                                                             00988000
22  %                                                                           00989000
23  STREAM PROCEDURE FINDOPTIONS(B,L);                                          00990000
24    BEGIN                                                                     00991000
25      LABEL L1,L2,L3,XIT;                                                     00992000
26      SI:=8; DI:=L; DI:=DI+8;                                                 00993000
27      DS:=16 LIT"MCP OPTIONS ARE:";                                           00994000
28      63(IF SC=" " THEN JUMP OUT TO L1 ELSE SI:=SI+1);                        00995000
29      GO XIT;                                                                 00996000
30    L1:                                                                       00997000
31      63(IF SC=" " THEN JUMP OUT TO L2 ELSE SI:=SI-1);                        00998000
32      GO XIT;                                                                 00999000
33    L2:                                                                       01000000
34      2(48(IF SC=ARROW THEN JUMP OUT 2 TO L3 ELSE DS:=CHR));                  01001000
35    L3: DS:=LIT",";                                                           01002000
36    XIT:                                                                      01003000
37    END FINDOPTIONS;                                                          01004000
38  %                                                                           01004050
39  BOOLEAN STREAM PROCEDURE XDBADISK(B,XDFILE,JUNK); VALUE XDFILE;             01004100
40  BEGIN                                                                       01004150
41      LOCAL BAD; LABEL L1,L2,L3,XIT;                                          01004200
42      SI:=B; IF SC≠" " THEN GO XIT;                                           01004250
43      SI:=SI+1; DI:=LOC BAD; DS:=6LIT"BADISK"; DI:=DI-6;                      01004300
44      IF 6 SC≠DC THEN GO XIT;                                                 01004350
45  L1: IF SC≠"/" THEN BEGIN SI:=SI+1; GO L1; END;                              01004400
46      SI:=SI+1; DI:=JUNK; DS:=LIT"C"; DS:=7CHR;                               01004450
47      SI:=SI-7; DS:=7OCT;                                                     01004500
48  L2: IF SC≠"=" THEN BEGIN SI:=SI+1; GO L2; END;                              01004550
49      SI:=SI+1; DS:=8 OCT;                                                    01004600
50      DS:=16LIT"0       ";                                                    01004650
51      XDFILE(DI:=DI-24; SKIP DB; DS:=SET; DI:=DI+8;                           01004700
52  L3: IF SC≠ARROW THEN BEGIN SI:=SI+1; GO L3; END;                            01004750
53      SI:=SI-15; DS:=7CHR;                                                    01004800
54      SI:=SI+1; DI:=DI+1; DS:=7CHR);                                          01004850
55      TALLY:=1;                                                               01004900
56  XIT: XDBADISK:=TALLY;                                                       01004950
57  END XDBADISK;                                                               01005000
```

```
%                                                                       01005050
      STREAM PROCEDURE SETEUSPEED(EUS,B,EUSPEED); VALUE EUS;            01006000
        BEGIN                                                           01007000
          SI:=B; SI:=SI+2; BI:=EUSPEED; SKIP DB;                        01008000
          EUS(IF SB THEN                                                01009000
              BEGIN DS:=SET; DS:=RESET; END ELSE                        01009100
              BEGIN DS:=RESET; DS:=SET; END; SKIP SB);                  01010000
        END SETEUSPEED;                                                 01011000
%                                                                       01011100
      STREAM PROCEDURE SPOUT(L,MFID,FID,DATE,RECS);                     01012000
        VALUE MFID,FID,DATE,RECS;                                       01013000
        BEGIN                                                           01014000
          DI:=L; SI:=LOC MFID; SI:=SI+1; DS:=LIT" ";                    01015000
          DS:=7 CHR; SI:=SI+1; DS:=LIT"/"; DS:=7 CHR;                   01016000
          DS:=12 LIT"  ACCESSED: "; DS:=8 CHR;                          01017000
          DS:=8 LIT"  SEGS: "; SI:=SI+4; DS:=4 CHR; DS:=LIT ARROW;      01018000
        END SPOUT;                                                      01019000
%                                                                       01020000
      STREAM PROCEDURE SCAN(A,ID);                                      01021000
        BEGIN LABEL L1,L2,L3,L4,XIT;                                    01022000
          SI:=A; DI:=ID; DS:=8LIT"OSTOP   "; DI:=DI-7;                  01022100
      L1: IF SC=" " THEN BEGIN SI:=SI+1; GO L1; END;                    01022200
          IF SC=ARROW THEN GO XIT;                                      01022300
          7(IF SC=ALPHA THEN DS:=CHR ELSE DS:=LIT" ");                  01022400
      L2: IF SC=ALPHA THEN BEGIN SI:=SI+1; GO L2; END;                  01022500
      L3: IF SC=" " THEN BEGIN SI:=SI+1; GO L3; END;                    01022600
          IF SC="/" THEN                                                01022700
          BEGIN                                                         01022800
            DS:=LIT"0";                                                 01022900
      L4:   SI:=SI+1;                                                   01023000
            IF SC=" " THEN GO L4;                                       01023100
            7(IF SC=ALPHA THEN DS:=CHR ELSE DS:=LIT" ");                01023200
          END;                                                          01023300
      XIT:                                                              01023400
          END SCAN;                                                     01030000
%                                                                       01031000
      INPASS1:=TRUE;                                                    01031100
      DRUMTOG:=SUMFILETOG:=CROSSTOG:=HALTOG:=XOTOG:=FALSE;              01031200
      NUMENTRIES:=ETYPE:=0;                                             01031300
      TAPERRORS:=DISKERRORS:=TAPETESTS:=0;                              01031400
FINDIT:                                                                 01032000
      TIME1:= TIME(1);                                                  01033000
      PROCTIME:= TIME(2);                                               01033100
      IOTIME:= TIME(3);                                                 01033200
      IF COMMON#2 THEN                                                  01034000
      BEGIN                                                             01035000
          WRITE(SPO[STOP],WFILE);                                       01036000
          READ(SPO,9,A[*]);                                             01037000
      END ELSE                                                          01038000
      BEGIN                                                             01038100
          READ(FINPUT,10,A[*]);                                         01038200
      END;                                                              01038300
      A[9]:= MARKER; FID:= LOGFID;                                      01039000
      SCAN(A,MFID);                                                     01040000
      IF MFID="STOP    " THEN GO TO EOP;                                01042000
      MAINTLOG:=(MFID="MAINT   ");                                      01042100
%                                                                       01042200
      CLOSE(LOG); % JUST TO BE SURE                                     01043000
      FILL LOG WITH MFID,FID;                                           01044000
      IF NOT USEFINPUT THEN                                             01045000
```

```
        BEGIN                                                                   01045100
        SEARCH(LOG,A[*]);                                                       01046000
        IF A[0]=(-1) THEN                                                       01047000
        BEGIN WRITE(SPO,NFILE,LMFID);                                           01048000
              GO FINDIT;                                                        01049000
        END;                                                                    01050000
        END USEFINPUT;                                                          01051000
%                                                                               01052000
        READ(LOG,5,A[*]);                                                       01057000
        SEGS:= A[0].[24:15]+2+REAL(MAINTLOG);                                   01058000
        %CROSSTOG:= NOT (A[0] < 0);                                             01059000
        IF COM=512 THEN                                                         01060000
        BEGIN SPOUT(L,MFID,FID,DAY(-A[4]),DECL(SEGS));                          01061000
              WRITE(SPO,10,L[*]); GO FINDIT;                                    01062000
        END;                                                                    01063000
        LASTENTRY:=A[0].[9:15];                                                 01063100
        STOPTIME:=A[1].[3:24];          % IN BINARY (1/60THS)                   01063200
        IF STOPTIME=0 THEN STOPTIME:=TIME1;                                     01063300
        LOGVERSION:=A[3];                                                       01063400
        MAXTYPE:=LOGVERSION.[30:6];                                             01063500
        LOGVERSION:=LOGVERSION.[42:6];                                          01063600
        STOPDATE:= A[4];                % IN BCL (YYDDD)                        01064000
        SPACE(LOG,5);                                                           01066000
        RCNT:= 5;                                                               01067000
        READ(LOG[NO],5,A[*]);                                                   01068000
        LASTENTRY:=LASTENTRY-A[0].[25:14]+1;                                    01068100
        STARTDATE:=CDATE:= DECM(A[1].[30:18]);                                  01069000
        STARTIME:= A[1].[3:24];                                                 01070000
REED:                                                                           01071000
        READ(LOG,5,A[*])[ENDOFILE];                                             01072000
        LOGENTRY:= RCNT:= RCNT+1;                                               01073000
        IF (TYPE:=A[0].[3:6]) = 0 THEN GO REED;                                 01074000
        IF A[0] = MARKER THEN GO EXIT;                                          01075000
%                                                                               01076000
        NUMENTRIES:= NUMENTRIES+1;                                              01076100
        I:=A[0].[39:9]*5;                                                       01076200
        IF I>MAXENTRY THEN GO INVALDINDEX;                                      01076300
        MOVE(5,A,B);                                                            01079000
        FOR J:=5 STEP 5 UNTIL I DO                                              01080000
        BEGIN READ(LOG,5,A[*])[ENDOFILE];                                       01081000
              RCNT:= RCNT+1;                                                    01082000
              MOVE(5,A,B[J]);                                                   01083000
        END;                                                                    01084000
        IF TYPE>MAXTYPE THEN GO REED;                                           01084100
        GO SWTYPE[TYPE];                                                        01085000
PERIP:                                                                          01086000
        IF LISTOG OR SUMTOG THEN PERIPHANDLER;                                  01087000
        GO CROSSREF;                                                            01088000
OBJOB:                                                                          01089000
        B[2].[3:15]:= LOGENTRY;                                                 01090000
        MOVE(3,B[2],JAR[MIX:=B[0].[20:5],0]);                                   01091000
        IF MIX>MIXMAX THEN MIXMAX:=MIX;                                         01092000
        GO CROSSREF;                                                            01093000
FILEN:                                                                          01094000
        B[2].[3:15]:= LOGENTRY;                                                 01095000
        IF (FNUM:=B[0].[9:9])>MAXFNUM THEN FNUM:=MAXFNUM;                       01095100
        MOVE(3,B[2],JAR[(MIX:=B[0].[20:5]),FNUM*3]);                            01096000
        LOGENTRY.[3:15]:= JAR[MIX,0].[3:15];                                    01097000
        GO CROSSREF;                                                            01098000
EOJOB:                                                                          01099000
```

```
        J:= JAR[MIX:=B[0].[20:5],0],[3:15];                                  01100000
        IF CROSSTOG THEN                                                    01100100
        FOR I:= 1 STEP 1 UNTIL INX[31] DO                                   01101000
        IF TABLE[31,I]=J THEN TABLE[31,I].[3:15] := LOGENTRY;               01102000
        LOGENTRY.[3:15] := J;                                               01103000
        ZERO(300,JAR[MIX,0]);                                               01104000
        GO CROSSREF;                                                        01105000
HALOD:                                                                      01106000
        IF LISTOG THEN                                                      01107000
        BEGIN                                                               01108000
        BLANK(L);                                                           01109000
        WRITEPAGE;                                                          01110000
        MOVEMESS(B[2],L);                                                   01111000
        WRITEDOUBLE;                                                        01112000
        WRITE(L[*],FTIME,ATIME(B[1].[3:24])); WRITEDOUBLE;                  01113000
        WRITE(L[*],FDATE                                                    01114000
                  ,CDATE:=DECM(B[1].[30:18])                                01115000
                  ,DAY(CDATE)                                               01116000
                  ,ACTDATE                                                  01117000
                 ); WRITEDOUBLE;                                            01118000
        HALTOG:= TRUE;                                                      01119000
        END;                                                                01120000
        GO CROSSREF;                                                        01121000
WMCP:                                                                       01122000
        N:=B[3].[43:5];                                                     01122100
        IF N>20 THEN N:=20; % MAXIMUM OF 20 ELS                             01122200
        EUSPEED:=N;                                                         01122300
        SETEUSPEED(N,B[3],EUSPEED);                                         01123000
        FOR I:=1 STEP 1 UNTIL MIXMAX DO                                     01124000
        ZERO(300,JAR[I,0]);                                                 01125000
        MIXMAX:= 0;                                                         01126000
        IF LISTOG THEN                                                      01127000
        BEGIN                                                               01128000
        BLANK(L);                                                           01129000
        IF NOT HALTOG THEN                                                  01130000
        BEGIN                                                               01131000
          WRITEPAGE;                                                        01132000
          MOVEMESS(B[4],L);                                                 01133000
          WRITEDOUBLE;                                                      01134000
          WRITE(L[*],FTIME,ATIME(B[1].[3:24])); WRITEDOUBLE;                01135000
          WRITE(L[*],FDATE                                                  01136000
                    ,CDATE:=DECM(B[1].[30:18])                              01137000
                    ,DAY(CDATE)                                             01138000
                    ,ACTDATE                                                01139000
                   ); WRITEDOUBLE;                                          01140000
        END ELSE                                                            01141000
        BEGIN FINDOPTIONS(B[4],L);                                          01142000
              WRITEDOUBLE;                                                  01143000
              HALTOG:= FALSE;                                               01144000
        END;                                                                01145000
        WRITE(L[*],WM,B[2]); WRITEDOUBLE;                                   01146000
        WRITE(L[*],NEU,N); WRITEDOUBLE;                                     01147000
        FOR I:=1 STEP 1 UNTIL N DO                                          01148000
        BEGIN                                                               01149000
        WRITE(L[*],MODEL                                                    01150000
                  ,I-1                                                      01151000
                  ,IF (J:=GETEUSPEED(I-1,EUSPEED))=2 THEN "1B" ELSE "1 "    01152000
                  ,J*20                                                     01153000
                 ); WRITEDOUBLE;                                            01154000
        END;                                                                01155000
```

```
        BLANK(L); WRITESPACES(4);                                          01156000
        END;                                                               01157000
        GO CROSSREF;                                                       01158000
TMDAT:                                                                     01159000
        IF B[3]≠0 THEN                                                     01160000
        IF (I:=DECM(B[1].[30:18]))≠CDATE THEN                              01161000
        BEGIN                                                              01162000
              CDATE:=I;                                                    01163000
              IF LISTOG THEN                                               01164000
              BEGIN                                                        01165000
              WRITEPAGE;                                                   01166000
              WRITE(L[*],FTIME,ATIME(B[1].[3:24])); WRITEDOUBLE;           01167000
              WRITE(L[*],FDATE                                             01168000
                        ,CDATE                                             01169000
                        ,DAY(CDATE)                                        01170000
                        ,ACTDATE                                           01171000
                       ); WRITEDOUBLE;                                     01172000
              END;                                                         01173000
        END;                                                               01174000
        GO CROSSREF;                                                       01175000
XDISK:                                                                     01176000
        IF LISTOG THEN                                                     01177000
        BEGIN                                                              01178000
        BLANK(L);                                                          01179000
        MOVEMESS(B[3],L[1]);                                               01180000
        L[0]:= "    " & ATIME(B[1].[3:24])[1:19:29];                       01181000
        WRITESPACES(3);                                                    01182000
        END;                                                               01183000
        IF SUMTOG THEN                                                     01183050
        BEGIN                                                              01183100
              IF XDBADISK(B[3],BL(B[2].[1:1]),JUNKBUFFER[2]) THEN          01183150
              BEGIN                                                        01183200
                    MOVE(2,B,JUNKBUFFER);                                  01183250
                    JUNKBUFFER[7]:=JUNKBUFFER[4]&JUNKBUFFER[3][8:23:25];   01183300
                    WRITE(BADISK,10,JUNKBUFFER[*]);                        01183350
                    XDTOG:=TRUE;                                           01183400
              END;                                                         01183450
        END;                                                               01183500
        GO CROSSREF;                                                       01184000
FECOM:                                                                     01185000
        IF LISTOG THEN                                                     01186000
        BEGIN                                                              01187000
        BLANK(L);                                                          01188000
        MOVEMESS(B[2],L[2]);                                               01189000
        L[0]:= "    " & ATIME(B[1].[3:24])[1:19:29];                       01190000
        L[1]:= "OMMENT " & 19[1:43:5];                                     01191000
        WRITESPACES(3);                                                    01192000
        END;                                                               01193000
        GO CROSSREF;                                                       01193100
TAPETST:                                                                   01193200
        IF TESTOG THEN                                                     01193300
        BEGIN                                                              01193400
          MOVE(3,JAR[MIX:=B[0].[20:5],0],B[200]);                          01193500
          IF (FNUM:=B[0].[9:9])>MAXFNUM THEN FNUM:=MAXFNUM;                01193550
          MOVE(3,JAR[MIX,FNUM×3],B[203]);                                  01193600
          B[206]:= CDATE;                                                  01193700
          WRITE(TST,210,B[*]); TAPETESTS:= TAPETESTS+1;                    01193800
        END;                                                               01193900
CROSSREF:                                                                  01194000
        IF CROSSTOG THEN                                                   01195000
```

```
          BEGIN                                                                 01196000
          IF TYPE < 12 THEN                                                     01197000
          BEGIN                                                                 01198000
1           IF MIX#0 THEN LOGENTRY:= LOGENTRY & JAR[MIX,0] [3:3:15];            01199000
2           IF FNUM#0 THEN LOGENTRY:= LOGENTRY & JAR[MIX,FNUM*3] [18:3:15];     01200000
3           TABLE[LUN,(INX[LUN]:= INX[LUN]+1)] := LOGENTRY;                     01201000
4         END ELSE                                                              01202000
5         BEGIN                                                                 01203000
6           TABLE[(TYPE:=TYPE+19),(INX[TYPE]:= INX[TYPE]+1)] := LOGENTRY;       01204000
7         END;END;                                                              01205000
8         GO REED;                                                              01205100
9   ONLTEST:                                                                    01205200
10        IF ONLTOG THEN                                                        01205300
11        BEGIN                                                                 01205400
12          IF TYPE:=B[0].[9:3] = 2 THEN                                        01205500
13          BEGIN                                                               01205600
14            MOVE(7,B,L);                                                      01205700
15            L[1]:= L[1] & OCTV(CDATE)[27:27:21];                              01205800
16            L[7]:= 0 & (LUN:=UNIT[B[3].[3:5]*2])[1:43:5];                     01205900
17            IF B[0].[18:2]=1 THEN %TAPE                                       01206000
18            BEGIN                                                             01206100
19              L[0]:= L[0] & B[1][27:27:21];                                   01206200
20              L[7]:= L[7] & B[7][15:30:18] &                                  01206300
21                                  LUN[33:42:6] & B[9][39:42:6];               01206400
22              L[8]:= TPECNF[0,MIX:=B[0].[20:5]];                              01206500
23              L[9]:= TPECNF[1,MIX];                                           01206600
24            END ELSE                                                          01206800
25            BEGIN                                                           01206900
26              BREAKDOWNADDRESS(B[2],A);                                     01207000
27              L[7]:= L[7] & REAL(A[9]<0)[7:47:1] &                          01207100
28                                  A[0][8:23:25] & B[3][33:27:6] & B[2][41:1:4];01207200
29            END;                                                            01207400
30            WRITE(TEMP,10,L[*]);                                            01207500
31            SUMFILETOG:= TRUE;                                              01207600
32          END ELSE                                                          01207700
33          IF TYPE=1 AND B[0].[18:2]=1 THEN                                  01207800
34          BEGIN TPECNF[0,MIX:=B[0].[20:5]]:= B[5];                          01207900
35                TPECNF[1,MIX]:= B[6];                                       01208000
36          END;                                                              01208100
37        END;                                                                01208200
38        GO CROSSREF;                                                        01208300
39  EXIT:                                                                     01208400
40  END PASS1;                                                                01208500
41  %                                                                         01226000
42  PROCEDURE PASS2;                                                          01227000
43  BEGIN LABEL CLEANUP;                                                      01228000
44        FORMAT                                                              01228100
45          WRAPUP(/////,"LOG FILE IS",X5,A1,A6,"/",A1,A6,                    01228200
46                 2(//,"LOG ",A5,X15,A5,X2,A6,"DAY    ",O,X3,A5),            01228300
47                   //,"LOG ENTRIES ",X4,I5," OF ",U,                        01228400
48                   //,"LOG SEGMENTS",X4,I5,                                 01228500
49                  ///,"PROCESSOR TIME   ",I4," MIN. ",F5.1," SEC.",        01228600
50                   //,"I/O TIME         ",I4," MIN. ",F5.1," SEC.",        01228700
51                   //,"ELAPSED TIME     ",I4," MIN. ",F5.1," SEC.");       01228800
52        INPASS1:=FALSE;                                                     01228900
53        IF SUMTOG THEN                                                      01229000
54        BEGIN                                                               01229050
55        IF SUMFILETOG THEN  % THERE ARE ERROR ENTRIES IN SUM FILE           01230000
56        BEGIN                                                               01231000
57        A[7]:= MARKER; WRITE(TEMP,10,A[*]);                                 01231100
```

```
                REWIND(TEMP);                                                   01231200
                J:= 0;                                                          01231300
                SORT(OUTSORT,INLUNSORT,0,HI,CMPLUN,10,6000);                    01231400
1               READ(SUM[NO]);                                                  01231500
2               GROUPTOTALS(0,TAPERRORS);                                       01232000
3               IF TAPERRORS≠0 THEN                                             01233000
4               BEGIN                                                           01234000
5                 REWIND(SUM); READ(SUM[NO]);                                   01235000
6                 J:= 1; LUNMAX:= 15;                                           01236000
7                 SORT(OUTSORT,INSORT,0,HI,CMP,10,6000);                        01237000
8                 REPORTPRN;                                                    01238000
9               END;                                                            01240000
10              GROUPTOTALS(1,DISKERRORS);                                      01242000
11              IF DISKERRORS≠C THEN                                            01243000
12              BEGIN                                                           01244000
13                SPACE(SUM,-DISKERRORS); READ(SUM[NO]);                        01245000
14                J:=1; LUNMAX:= 19;                                            01246000
15                SORT(OUTSORT,INSORT,0,HI,CMP,10,6000);                        01247000
16                REPORTADDRESS;                                                01248000
17              END;                                                            01250000
18              END SUMFILETOG;                                                 01250050
19              IF XDTOG THEN                                                   01250100
20              BEGIN                                                           01250150
21                A[7]:=MARKER; WRITE(BADISK,10,A[*]);                          01250200
22                REWIND(BADISK);                                               01250250
23                J:=1;                                                         01250300
24                SORT(OUTSORT,INLUNSORT,0,HI,CMP,10,6000);                     01250350
25                REPORTBADISK;                                                 01250400
26              END XDTOG;                                                      01250450
27              IF SUMFILETOG THEN                                              01250500
28              BEGIN                                                           01250550
29              IF DRUMTOG THEN BEGIN REWIND(SUM); READ(SUM[NO]); END;          01251000
30              FOR J:=3-REAL(DRUMTOG) STEP 1 UNTIL 8 DO GROUPTOTALS(J,T1);     01252000
31              END;                                                            01252050
32              END SUMTOG;                                                     01252100
33              IF TESTOG THEN IF TAPETESTS≠0 THEN REPORTAPETEST;               01252200
34              IF LISTOG THEN                                                  01253000
35              BEGIN                                                           01254000
36                IF ETYPE≠0 THEN                                               01254100
37                BEGIN                                                         01254200
38                  BLANK(L); WRITEDOUBLE;                                      01254300
39                  MOVE(10,TERMINALMESSAGA,L);                                 01254400
40                  WRITEDOUBLE;                                                01254500
41                END;                                                          01254600
42                L[17]:=1023; WRITE(PBD,18,L[*]);                              01255000
43                REWIND(PBD);                                                  01256000
44                WHILE TRUE DO                                                 01257000
45                BEGIN                                                         01258000
46                  READ(PBD,18,L[*]);                                          01259000
47                  IF EOFPBD THEN GO CLEANUP;                                  01260000
48                  IF PAGEIT THEN                                              01261000
49                  BEGIN                                                       01261010
50                    WRITE(LPA[PAGE]) ;                                        01261020
51                    PRINTMFID;                                                01261030
52                  END ELSE                                                    01261040
53                  IF SINGLESPACE THEN WRITE(LPA,15,L[*]) ELSE                 01262000
54                  IF DOUBLESPACE THEN WRITE(LPA[DBL],15,L[*]) ELSE            01263000
55                  BEGIN WRITE(LPA,15,L[*]);                                   01264000
56                        WRITE(LPA,SPACES,L[17]);                              01265000
57                  END;                                                        01266000
```

```
          END;                                                                  01267000
CLEANUP:                                                                        01268000
          REWIND(PBD);                                                          01269000
        END LISTOG;                                                             01270000
%                                                                               01270100
        IF SUMFILETOG THEN REWIND(SUM);                                         01271000
        WRITE(LPA[PAGE]);                                                       01275000
        PRINTMFID ;                                                             01275010
        BLANK(L);                                                               01276000
        BOJMESS(L,TIMEO,TIME(6),DAY(-TIMEO),ATIME(TIME1),                       01277000
                LISTOG,SUMTOG,TESTOG,ONLTOG);                                   01278000
        WRITE(LPA[DBL],15,L[*]);                                                01279000
        IF ETYPE#0 THEN                                                         01279100
        BEGIN                                                                   01279200
          WRITE(LPA[DBL]); WRITE(LPA,10,TERMINALMESSAGA[*]);                    01279300
        END;                                                                    01279400
        IF LASTENTRY<NUMENTRIES OR MAINTLOG THEN                                01279500
        LASTENTRY:=NUMENTRIES;                                                  01279600
        WRITE(LPA,WRAPUP                                                        01280900
                ,MFID.[6:6],MFID,FID.[6:6],FID                                  01281000
                ,"START",STARTDATE,DAY(STARTDATE),ACTDATE,ATIME(STARTTIME)      01282000
                ,"END  ",STOPDATE,DAY(STOPDATE),ACTDATE,ATIME(STOPTIME)         01283000
                ,NUMENTRIES,LASTENTRY,SEGS                                      01284000
                ,T1:=(T2:= TIME(2)-PRUCTIME) DIV 3600                           01285000
                ,(T2/60) - T1*60                                                01286000
                ,T1:=(T2:= TIME(3)-IOTIME) DIV 3600                             01287000
                ,(T2/60) - T1*60                                                01288000
                ,T1:=(T2:= TIME(1)-TIME1) DIV 3600                              01289000
                ,(T2/60) - T1*60                                                01290000
                );                                                              01291000
      END PASS2;                                                                01293000
%                                                                               01294000
% START OF OUTER BLOCK %                                                        01295000
%                                                                               01296000
      INITIALIZE;                                                               01297000
STARTUP:                                                                        01298000
      PASS1;                                                                    01299000
FINISHUP:                                                                       01300000
      PASS2;                                                                    01301000
      GO STARTUP;                                                               01302000
%                                                                               01303000
FLAGBIT:                                                                        01304000
      ETYPE:=1;                                                                 01305000
      GO RENTER;                                                                01306000
INVALDINDEX:                                                                    01307000
      ETYPE:=2;                                                                 01308000
      GO RENTER;                                                                01309000
INTEGROVFLW:                                                                    01310000
      ETYPE:=3;                                                                 01311000
      GO RENTER;                                                                01312000
EXPONOVFLW:                                                                     01313000
      ETYPE:=4;                                                                 01314000
      GO RENTER;                                                                01315000
DIVBYZERO:                                                                      01316000
      ETYPE:=5;                                                                 01317000
      GO RENTER;                                                                01318000
ENDOFILE:                                                                       01319000
      ETYPE:=6;                                                                 01320000
RENTER:                                                                         01321000
      I:=ETYPE*2;                                                               01322000
```

```
        ALPHA1:=MESS[12+I]; ALPHA2:=MESS[13+I];                    01323000
        WRITE(TERMINALMESSAGA[*],FTERMINATE,LALPHA);               01324000
        WRITE(SPO,10,TERMINALMESSAGA[*]);                          01325000
        IF INPASS1 THEN GO FINISHUP;                               01326000
        GO STARTUP;                                                01327000
    EOP:                                                           01328000
    FINIS:                                                         01329000
    END PROGRAM.                                                   01330000
    END;END.        LAST CARD ON OCRDING TAPE                      99999999
```

LABEL 000000000PRINTER001/5100CC LX OBJECT/READ;FILE SOURCEFILE=SYMBOL/MLUGAN←←0000000 OBJECT /READ

Data Documents/Inc.