

LABEL 000000000PRINTER00175100CC EX OBJECT/READ;FILE SOURCEFILE=SYMBOL/AUXDATA\*0000000

OBJECT /READ

SYMBOL/AUXDATA

Data Documents/Inc.

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57

4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57

```

BEGIN
% AUXMEM UTILITY PROGRAM - OCTOBER, 1971
COMMENT: * TITLE: B5500/B5700 MARK XIV SYSTEM RELEASE *
1 * FILE ID: SYMBOL/AUXDATA TAPE ID: SYMBOL2/FILE000 *
2 * THIS MATERIAL IS PROPRIETARY TO BURROUGHS CORPORATION *
3 * AND IS NOT TO BE REPRODUCED, USED, OR DISCLOSED *
4 * EXCEPT IN ACCORDANCE WITH PROGRAM LICENSE OR UPON *
5 * WRITTEN AUTHORIZATION OF THE PATENT DIVISION OF *
6 * BURROUGHS CORPORATION, DETROIT, MICHIGAN 48232 *
7 *
8 * COPYRIGHT (C) 1972 BURROUGHS CORPORATION *
9 * AA386657 *;
10 % DOES THE FOLLOWING BASED ON COMMON VALUE:
11 % COMMON=0:
12 % CREATES "AUXMCP/DISK" FILE FROM STUFF DECK AND PROCEDURE NAMES
13 % COMMON=1:
14 % CREATES "AUXINT/DISK" FILE FROM INTRINSIC NUMBERS
15 % COMMON=2:
16 % LISTS OCCURRENCES OF MCP AND INTRINSIC DISK ACCESSES
17 % COMMON=3:
18 % TRACES OCCURRENCES OF MCP FUNCTIONS
19 % COMMON=4:
20 % PUNCHES MCP PROCEDURE NAMES FROM STUFF DECK
21 REAL COMMON;
22 FILE CARD 0(2,10,30);
23 FILE PUNCH 0(2,10,30);
24 FILE LINE 1(2,15);
25 FILE SPO 11(2,10);
26 FILE STUFF DISK SERIAL (2,10,30);
27 FILE MCPFILE DISK SERIAL [20:30] "AUXMCP ""DISK "" (2,30,SAVE 30);
28 FILE INTFILE DISK SERIAL [20:30] "AUXINT ""DISK "" (2,30,SAVE 30);
29 FILE SYSMTR DISK SERIAL "SYSTEM ""MONITOR" (2,30);
30 DEFINE NUMOFSTUFROWS = 4#, STUFROWSIZE=512#;
31 DEFINE MATCH(MATCH1,MATCH2)=
32 (REAL(BOOLEAN(MATCH1) EQV BOOLEAN(MATCH2))=REAL(NOT FALSE))#;
33 REAL I,K,PTR,SPTR,CELL,CELLCOUNT,MAXROWS,MINCELL,MAXCELL;
34 ARRAY STUF,KEYS[0:NUMOFSTUFROWS,0:STUFROWSIZE];
35 ARRAY TEMP[0:15],A[0:30];
36 SAVE ARRAY CRD[0:10];
37 %*****00012900
38 STREAM PROCEDURE MOVE(N,A,B); VALUE N;
39 BEGIN SI:=A; DI:=B; DS:=N WDS; END;
40 %*****00013200
41 STREAM PROCEDURE ZERO(A);
42 BEGIN DI:=A; DS:=8LIT"0"; SI:=A; DS:=29 WDS; END;
43 %*****00013500
44 REAL STREAM PROCEDURE ADDRESS(A);
45 BEGIN SI:=A; ADDRESS:=SI; END;
46 %*****00013800
47 DEFINE PRINTCELL=
48 SI:=LOC CELL; SV1:=DI; DS:=8DEC; DS:=LIT"%"; DI:=SV1; DS:=7FILL;
49 SV2:=DI; SI:=SV2; DI:=SV1;
50 9(IF SC="%" THEN JUMP OUT ELSE DS:=CHR);
51 DS:=3LIT" ("; SV1:=DI; SI:=LOC CELL;
52 16(DS:=3RESET; 3(IF SB THEN DS:=SET ELSE DS:=RESET; SKIP SB));
53 DS:=LIT"%"; DI:=SV1; DS:=15FILL; SV2:=DI; SI:=SV2; DI:=SV1;
54 17(IF SC="%" THEN JUMP OUT ELSE DS:=CHR); DS:=2LIT") "#;
55 %*****00014700
56 STREAM PROCEDURE OUTFMT(SGN,CELL,TYPE,N,NFLG,CRD,TEMP,SWTCH);
57 %*****00014900

```

Data Documents/Inc.

33523

	VALUE SGN,CELL,TYPE,N,NFLG,SWTCH;	00015000
	% FORMATS INFORMATION FOR OUTPUT TO PRINTER	00015100
	% "SGN " IS VALUE IN [1:1] FIELD OF SYSTEM/MONITOR WORD	00015200
1	% "CELL" IS VALUE IN [9:39] FIELD OF SYSTEM/MONITOR WORD	00015300
2	% "TYPE" IS VALUE IN [3:6] FIELD OF SYSTEM/MONITOR WORD	00015400
3	% " N " IS NUMBER OF OCCURANCES OF CELL.[9:39] IN MONITOR FILE	00015500
4	% "NFLG" IS TRUE IF PRINTING THE NUMBER OF OCCURANCES	00015600
5	% "CRD " IS STUFF FILE CARD IMAGE RECORD	00015700
6	% "TEMP" IS OUTPUT ARRAY TO BE PRINTED	00015800
7	% "SWTCH" (VALUES 0,1,2 OR 3) DEFINES FORMAT OF PRINTED RECORDS	00015900
8	BEGIN LABEL L1,L2,L3,L4,CT,EXIT; LOCAL SV1,SV2;	00016000
9	DI:=TEMP; DS:=8LIT" "; SI:=TEMP; DS:=14 WDS; DI:=TEMP;	00016100
10	CI:=CI+SWTCH;	00016200
11	GO TO L1; % MCP PROCEDURE NAME	00016300
12	GO TO L2; % INTRINSIC NUMBER	00016400
13	GO TO L3; % TRACE (SPECIAL) NUMBER	00016500
14	GO TO L4; % RESTART MARKER	00016600
15	L1: DS:=9LIT"PRT CELL "; PRINTCELL;	00016700
16	SI:=CRD; SI:=SI+8; 63(IF SC=ALPHA THEN DS:=CHR ELSE JUMP OUT);	00016800
17	NFLG(JUMP OUT TO CT); GO TO EXIT;	00016900
18	L2: DS:=10LIT"INTRINSIC "; PRINTCELL;	00017000
19	NFLG(JUMP OUT TO CT); GO TO EXIT;	00017100
20	L3: DS:=6LIT"TRACE "; SI:=LOC TYPE; DS:=2DEC; DS:=3LIT" = ";	00017200
21	SGN(DS:=LIT"="); PRINTCELL;	00017300
22	NFLG(JUMP OUT TO CT); GO TO EXIT;	00017400
23	L4: DS:=10LIT"*"; DS:=13LIT"RESTART POINT"; DS:=10LIT"*"; GO EXIT;	00017500
24	CT: DS:=10LIT" OCCURRED "; SI:=LOC N; SV1:=DI; DS:=RDEC; DS:=LIT"%";	00017600
25	DI:=SV1; DS:=7FILL; SV2:=DI; SI:=SV2; DI:=SV1;	00017700
26	9(IF SC="%" THEN JUMP OUT ELSE DS:=CHR);	00017800
27	DS:=7LIT" TIMES";	00017900
28	EXIT: DS:=16LIT" ";	00018000
29	END STREAM PROCEDURE OUTFMT;	00018100
30	*****00018200	
31	PROCEDURE SETKEYS;	00018300
32	*****00018400	
33	% BUILD KEYS TO STUFF FILE FOR SUBSEQUENT RANDOM READS FROM DISK	00018500
34	BEGIN LABEL EOF; REAL CELL,DISKCOUNT; INTEGER ROW,COL;	00018600
35	REAL STREAM PROCEDURE CELLNUMBER(CRD);	00018700
36	% OBTAINS PRT CELL NUMBER FROM STUFF DECK CARD IMAGE	00018800
37	BEGIN	00018900
38	SI:=CRD; SI:=SI+4; DI:=LOC CELLNUMBER; DS:=40CT;	00019000
39	END STREAM PROCEDURE CELLNUMBER;	00019100
40	DISKCOUNT:=-1; MINCELL:=MAXCELL:=0;	00019200
41	DO BEGIN	00019300
42	READ(STUFF,10,CRD[*])(EOF); DISKCOUNT := DISKCOUNT + 1;	00019400
43	CELL := CELLNUMBER(CRD);	00019500
44	IF CELL LSS MINCELL THEN MINCELL:=CELL;	00019600
45	IF CELL GTR MAXCELL THEN MAXCELL:=CELL;	00019700
46	ROW := CELL DIV STUFROWSIZE; COL:=CELL MOD STUFROWSIZE;	00019800
47	KEYS(ROW,COL):=DISKCOUNT;	00019900
48	END UNTIL FALSE;	00020000
49	EOF; END PROCEDURE SETKEYS;	00020100
50	*****00020200	
51	CASE COMMON OF	00020300
52	BEGIN	00020400
53	BEGIN % CASE 0	00020500
54	LABEL READCARD,ERROR,EOF,EXIT;	00020600
55	FORMAT CARDERROR("ERROR IN THIS CARD"),	00020700
56	NOSTUFDECK("NO INFORMATION IN STUFF DECK");	00020800
57	% SET UP FILE AUXMCP/DISK USING STUFF FILE AND PROCEDURE NAMES	00020900

Data Documents/Inc.

33522

```

% AS SPECIFIED BY USER                                00021000
%*****00021100
BOOLEAN STREAM PROCEDURE SCANCARD(CRD,TEMP);          00021200
%*****00021300
% TRANSFERS PROCEDURE NAME FROM CARD IMAGE TO "TEMP[0]" IN A7 FMT 00021400
% TRANSFERS CHARACTER SIZE TO "TEMP[0]"              00021500
BEGIN LOCAL T; LABEL EXIT;                            00021600
SI:=CRD;                                               00021700
20(IF SC=ALPHA THEN IF SC LSS "0" THEN JUMP OUT; SI:=SI+1); 00021800
IF TOGGLE THEN ELSE                                  00021900
BEGIN                                                 00022000
TALLY:=1; SCANCARD:=TALLY; GO TO EXIT;                00022100
END;                                                  00022200
DI:=TEMP; DI:=DI+8;                                   00022300
9(DS:=8LIT"C "; DI:=DI-7;                             00022400
7(IF SC=ALPHA THEN ELSE JUMP OUT 2 TO EXIT;           00022500
DS:=CHR; TALLY:=TALLY+1));                            00022600
EXIT: T:=TALLY; SI:=LOC T; DI:=TEMP; DS:=WDS;         00022700
END STREAM PROCEDURE SCANCARD;                        00022800
%*****00022900
BOOLEAN STREAM PROCEDURE SCANSTUFF(CRD,TEMP,SIZE,CELL); 00023000
%*****00023100
% TRANSFERS PROCEDURE NAME TO TEMP IN A7 FORMAT FROM STUFF DECK 00023200
% TRANSFERS LENGTH OF NAME AND PRICELL NUMBER TO SIZE AND CELL 00023300
BEGIN LOCAL T; LABEL L1,EXIT;                         00023400
SI:=CRD; SI:=SI+2;                                    00023500
IF SC NEQ "1" THEN % NOT PROCEDURE NAME              00023600
BEGIN                                                 00023700
TALLY:=1; SCANSTUFF:=TALLY; GO TO EXIT;              00023800
END;                                                  00023900
SI:=SI+2; DI:=CELL; DS:=40CT; DI:=TEMP;             00024000
9(DS:=8LIT"C "; DI:=DI-7;                             00024100
7(IF SC=ALPHA THEN ELSE JUMP OUT 2 TO L1;            00024200
DS:=CHR; TALLY:=TALLY+1));                            00024300
L1: T:=TALLY; SI:=LOC T; DI:=SIZE; DS:=WDS;          00024400
EXIT: END STREAM PROCEDURE SCANSTUFF;                 00024500
%*****00024600
BOOLEAN PROCEDURE SETUPSTUFARRAY;                     00024700
%*****00024800
BEGIN                                                 00024900
% TRANSFER NAMES AND CELL NO. FROM STUFF FILE TO "STUF" ARRAY 00025000
REAL I,J,JPREV,SIZE;                                  00025100
LABEL READIN,EOF,FINISH;                              00025200
FORMAT FMT("INSUFFICIENT TABLE SPACE FOR STUFF DECK"); 00025300
READIN: READ(STUFF,10,CRD[*])(EOF);                  00025400
IF SCANSTUFF(CRD,TEMP,SIZE,CELL) THEN GO READIN; %NOT PROCEDURE 00025500
SIZE := (SIZE + 6 ) DIV 7;                             00025600
IF (SIZE+J) GTR STUFROWSIZE THEN % NO ROOM IN ROW    00025700
BEGIN                                                 00025800
STUF[I,JPREV].[17:11]:=1; % END OF ROW MARKER        00025900
IF (I=I+1) GTR NUMOFSTUFROWS THEN % NO MORE ROWS    00026000
BEGIN                                                 00026100
WRITE(SPO,FMT); GO TO FINISH;                        00026200
END;                                                  00026300
J:=0;                                                 00026400
END;                                                  00026500
STUF[I,J]:=SIZE & CELL[18:33:15];                    00026600
MOVE(SIZE,TEMP,STUF[I,J+1]);                          00026700
JPREV:=J;                                              00026800
J:=J+SIZE+1;                                          00026900

```

```

1      GO TO READIN;                                00027000
2      EOF:  IF J GTR STUFROWSIZE THEN * NO ROOM IN ROW 00027100
3            BEGIN                                  00027200
4            STUF[I,JPREV],[17:1]:=1; % END OF ROW MARKER 00027300
5            IF (I:=I+1) GTR NUMOFSTUFROWS THEN      00027400
6              BEGIN                                  00027500
7                WRITE(SPU,FMT); GO TO FINISH;      00027600
8              END;                                    00027700
9              J:=0;                                   00027800
10             END;                                    00027900
11             STUF[I,J]:=REAL(NOT FALSE);          00028000
12             MAXROWS := I;                          00028100
13             SETUPSTUFARRAY := STUF[0,0] NEQ REAL(NOT FALSE); 00028200
14             FINISH: END PROCEDURE SETUPSTUFARRAY;  00028300
15             %*****00028400
16             REAL PROCEDURE PRCELL(STUF,TEMP); ARRAY STUF,TEMP[0]; 00028500
17             %*****00028600
18             % COMPARES PROCEDURE NAMES FROM STUF DECK WITH CARD IMAGE NAME 00028700
19             % RETURNS PRCELL NUMBER OF PROCEDURE, IF FOUND; 00028800
20             BEGIN                                  00028900
21             REAL I,K,L,IMAX;                        00029000
22             BOOLEAN LASTENTRY;                     00029100
23             LABEL NEXTWORD,NEXT,EXIT;              00029200
24             NEXTWORD: IF STUF[I]=REAL(NOT FALSE) THEN GO TO EXIT; % END OF TABLE 00029300
25             IMAX := I + STUF[I],[33:15];           00029400
26             LASTENTRY := BOOLEAN(STUF[I],[17:1]); % END OF ROW MARKER 00029500
27             L:=0;                                   00029600
28             FOR K:=I+1 STEP 1 UNTIL IMAX DO         00029700
29               BEGIN                                  00029800
30                 IF ((TEMP[0]+6)DIV 7) NEQ (IMAX-I) THEN GO NEXT; % SIZE 00029900
31                 L:=L+1;                              00030000
32                 IF NOT MATCH(STUF[K],TEMP[L]) THEN  00030100
33                   BEGIN                                  00030200
34                     NEXT: IF LASTENTRY THEN GO TO EXIT; 00030300
35                     I:=IMAX + 1;                    00030400
36                     GO TO NEXTWORD;                 00030500
37                   END;                                00030600
38                 END;                                  00030700
39                 PRCELL:=STUF[I],[18:15];          00030800
40             END;                                    00030900
41             EXIT:  END PROCEDURE PRCELL;           00031000
42             %*****00031100
43             IF NOT SETUPSTUFARRAY THEN             00031200
44               BEGIN                                  00031300
45                 WRITE(SPU,NCSTUFDECK); GO TO EXIT;  00031400
46               END;                                    00031500
47             K:=1; ZERO(A);                           00031600
48             READCARD: READ(CARD,10,CRD[*])[EOF];   00031700
49             IF SCANCARD(CRD,TEMP) THEN GO TO ERROR; 00031800
50             FOR I:=0 STEP 1 UNTIL MAXROWS DO       00031900
51               IF (CELL:=PRCELL(STUF[I,*],TEMP)) GTR 0 THEN % FOUND MATCH 00032000
52                 BEGIN                                  00032100
53                   IF (K:=K+1) GTR 29 THEN % FILLED SEGMENT 00032200
54                     BEGIN                                  00032300
55                       WRITE(MCPFILE,30,A[*]);      00032400
56                       K:=0; ZERO(A);                00032500
57                     END;                                00032600
58                   A[K]:=CELL;                        00032700
59                   CELLCOUNT := CELLCOUNT + 1;      00032800
60                   GO TO READCARD;                   00032900
61                 END;

```

Data Documents/Inc





```

READ(STUFF[KEYS[I,J]],10,TEMP[*]); % STUFF DECK RECORD 00045000
MOVE(8,TEMP[0],CRD[2]); 00045100
END; % IF MCP PRT CELL VALUE 00045200
1 WRITE(SORTT,10,CRD[*]); FILEMAX:=FILEMAX+1; 00045300
2 END LOOP; 00045400
3 REWIND(SORTT); IF FILEMAX=0 THEN GO TO EXIT; 00045500
4 FCNT:=0; SORT(OUTPUT,INPUT,0,LOWVALUE,COMPARE,10,4000);FCNT:=0; 00045600
5 DO BEGIN 00045700
6 READ(SORTT,10,CRD[*]); 00045800
7 OUTFMT(0,CRD[1],TYPE,CRD[0],1,CRD[2],TEMP,TYPE); 00045900
8 WRITE(LINE[DBL],15,TEMP[*]); 00046000
9 END UNTIL (FCNT:=FCNT+1)=FILEMAX; 00046100
10 REWIND(SORTT); 00046200
11 EXIT: END PROCEDURE LISTOCCURANCES; 00046300
12 ***** 00046400
13 WRITE(LINE,HEADING1); SETKEYS; LISTOCCURANCES(0); 00046500
14 WRITE(LINE[DBL],HALTS,HL); WRITE(LINE[PAGE]); 00046600
15 FOR I:=0 STEP 1 UNTIL NUMOFSTUFROWS DO 00046700
16 FOR J:=0 STEP 1 UNTIL STUFROWSIZE DO STUF[I,J]:=0; 00046800
17 WRITE(LINE,HEADING2); LISTOCCURANCES(1); 00046900
18 END CASE 2; 00047000
19 %..... 00047100
20 BEGIN % CASE 3 ANALYZE SYSTEM/MONITOR FILE IN ORDER OF OCCURANCE 00047200
21 ***** 00047300
22 STREAM PROCEDURE OCTALDISPLAY(A,TEMP); 00047400
23 ***** 00047500
24 % CONVERTS SYSTEM/MONITOR REGRDLS TO OCTAL DIGITS FOR DISPLAY 00047600
25 BEGIN 00047700
26 SI:=A; DI:=TEMP; 00047800
27 6(2(8(DS:=3RESE1; 00047900
28 3(IF SB THEN DS:=SET ELSE DS:=RESE1; SK1P SB)); 00048000
29 DS:=LIT" "); DS:=2LIT" "); 00048100
30 END PROCEDURE OCTALDISPLAY; 00048200
31 ***** 00048300
32 STREAM PROCEDURE DASH(TEMP); 00048400
33 BEGIN DI:=TEMP; DS:=8LIT"-"; SI:=TEMP; DS:=14 WDS; END; 00048500
34 ***** 00048600
35 PROCEDURE OCTALTRACE; 00048700
36 ***** 00048800
37 % PRINTS SYSTEM/MONITOR RECORDS IN OCTAL FURM 00048900
38 BEGIN LABEL EOF; INTEGER I; 00049000
39 FORMAT FMT 00049100
40 (40("*"),"SYSTEM/MONITOR TRACE (OCTAL BY SEGMENT)",40("*"))//); 00049200
41 WRITE(LINE[PAGE]); WRITE(LINE,FMT); 00049300
42 DO BEGIN 00049400
43 DASH(TEMP); WRITE(LINE[DBL],15,TEMP[*]); 00049500
44 READ(SYSMTR,30,A[*])[EOF]; 00049600
45 IF A[0] = REAL(NOT FALSE) THEN GO TO EOF; 00049700
46 FOR I:=0 STEP 6 UNTIL 29 DO 00049800
47 BEGIN 00049900
48 OCTALDISPLAY(A[I],TEMP); WRITE(LINE[DBL],15,TEMP[*]); 00050000
49 END; 00050100
50 END UNTIL FALSE; 00050200
51 EOF: REWIND(SYSMTR); END PROCEDURE OCTALTRACE; 00050300
52 ***** 00050400
53 PROCEDURE TRACE; 00050500
54 ***** 00050600
55 % PRINTS VALUES FROM SYSTEM/MONITOR FILE IN ORDER OF OCCURANCE 00050700
56 BEGIN LABEL READIN,EOF; REAL CELL,N,SWTCH,TYPE,LOC,SGN; 00050800
57 FORMAT FMT(50("*"),"SYSTEM/MONITOR TRACE",50("*"))//); 00050900

```

Data Documents/Inc.

33520



```
WRITE(LINE[DBL],FMT); 00051000
DO BEGIN 00051100
READIN: READ(SYSMTR,30,A[+])(EOF); 00051200
        IF A[0] = REAL(NOT FALSE) THEN GO TO EOF; 00051300
        FOR I:=0 STEP 1 UNTIL 29 DO 00051400
        BEGIN 00051500
            CELL:=A[I].[9:39]; SGN:=A[I].[1:1]; TYPE:=A[I].[3:6]; 00051600
            IF TYPE=63 THEN GO TO READIN; % END OF ROW MARKER 00051700
            SWITCH:= 00051800
            IF TYPE=62 THEN 3 ELSE IF TYPE GTR 1 THEN 2 ELSE TYPE; 00051900
            IF TYPE=0 THEN % PRT CELL FOR MCP 00052000
            IF CELL GEQ MINCELL AND CELL LEQ MAXCELL THEN 00052100
            BEGIN 00052200
                LOC:=KEYS[CELL DIV STUFROWSIZE,CELL MOD STUFROWSIZE]; 00052300
                READ(STUFF[LOC],10,CRD[*]); 00052400
                END ELSE SWITCH:=2; % TREAT AS "TRACE" IF OUT OF RANGE 00052500
                OUTFMT(SGN,CELL,TYPE,0,0,CRD,TEMP,SWTCH); 00052600
                WRITE(LINE[DBL],15,TEMP[*]); 00052700
            END I LOOP; 00052800
        END UNTIL FALSE; 00052900
EOF: REWIND(SYSMTR); END PROCEDURE TRACE; 00053000
        SETKEYS; TRACE; OCTALTRACE; 00053100
        END CASE 3; 00053200
        ..... 00053300
        BEGIN % CASE 4 - PUNCH MCP PROCEDURE NAMES FROM STUFF DECK 00053400
        LABEL READIN,EOF; 00053500
        STREAM PROCEDURE REFORMAT(CRD,TEMP); 00053600
        BEGIN 00053700
            DI:=TEMP; DS:=@LIT" "; SI:=TEMP; DS:=9 WDS; 00053800
            SI:=CRD; SI:=SI+8; DI:=TEMP; 00053900
            63(IF SC=ALPHA THEN DS:=CHR ELSE JUMP OUT); 00054000
            END STREAM PROCEDURE REFORMAT; 00054100
            DO BEGIN 00054200
                READIN: READ(STUFF,10,CRD[*])(EOF); 00054300
                IF CRD[0].[12:6] NEQ "1" THEN GO TO READIN; % NOT PROCEDURE 00054400
                REFORMAT(CRD,TEMP); 00054500
                WRITE(PUNCH,10,TEMP[*]); 00054600
                END UNTIL FALSE; 00054700
            EOF: 00054800
            END CASE 4; 00054900
            END ALL CASES; 00055000
            END PROGRAM. 00055100
        END;END. LAST CARD ON OCRDING TAPE 99999999
        Q2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2Sx 00055200
        Q2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2SxQ2Sx 00055300
```

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
571  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57

LABEL 000000000PRINTER00175100CC EX OBJECT/READ;FILE SOURCEFILE=SYMBOL/AUXDATA+0000000

OBJECT /READ

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57

Data Documents/Inc.

33519