

SYMBOL/MERG

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

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.

```

BEGIN
  *** MERGE/CANDE SOURCE PROGRAM, 6-70. ***
  COMMENT: * TITLE: B5500/B5700 MARK XIV SYSTEM RELEASE
           * FILE ID: SYMBOL/MERG TAPE ID: SYMBOL2/FILE000
           * THIS MATERIAL IS PROPRIETARY TO BURROUGHS CORPORATION
           * AND IS NOT TO BE REPRODUCED, USED, OR DISCLOSED
           * EXCEPT IN ACCORDANCE WITH PROGRAM LICENSE OR UPON
           * WRITTEN AUTHORIZATION OF THE PATENT DIVISION OF
           * BURROUGHS CORPORATION, DETROIT, MICHIGAN 48232
           *
           * COPYRIGHT (C) 1971, 1972 BURROUGHS CORPORATION
           * AA320206 AA386657
  REAL COMMON;
  SAVE ARRAY ERR(0:4), A, N(0:30), B, PARAMS(0:10);
  FILE IN SOURCEFIL DISK SERIAL (2,0,0);
  FILE IN MERGFIL DISK SERIAL (2,0,0);
  FILE OUT OUTFIL DISK SERIAL [20:600] (2,10,300,SAVE 1);
  FILE OUT NEWTAB DISK SERIAL [20:30 ] (2,30,300,SAVE 1);
  BOOLEAN BRAAK, DELTOG, MERGREAD, RESEQ, KEVTOG, SEQCHK, SFLG, SUPPRESS, OK;
  BOOLEAN MAKTAB;
  LABEL EMPTY, FULL, NEXTPARAM, TESTREAD, READMERG, HAVEMERG,
  MERGFILES, EOFSOURCE, EOFMERG, EXIT;
  REAL LASTMERGE;
  REAL ADJ, BASE, EOFPIR, ENDELETE, ENDRESEQ, EOFMARK, HRANGE,
  INCR, LINE, LRANGE, LREC, MERGESEQ, NCT, NMG, NPARAMS, NPTR,
  OUTSEQ, PC, PREVSEQ, REC, SOURCESEQ, STRDELETE, STRRESEQ, USER;
  *****
  REAL STREAM PROCEDURE HDR(F,N); VALUE N;
  *****
  BEGIN
    SI:=F; 3(SI:=SI-8); DI:=LOC F; DS:=WDS;
    SI:=F; 14(SI:=SI+8); DI:=LOC F; DS:=WDS;
    SI:=F; N(SI:=SI+8); DI:=LOC HDR; DS:=WDS;
  END STREAM PROCEDURE HDR;
  *****
  PROCEDURE DISKWAIT(I,A,S,D);
  VALUE I,S,D; REAL I,S,D; ARRAY A[*]; COMMUNICATE(-8);
  *****
  PROCEDURE TWXOUT(A,N,T);
  *****
  VALUE N,T;
  REAL A,N,T;
  BEGIN COMMUNICATE(-11);
  BRAAK := BOOLEAN(T); % MCP RETURNS 1 IF BREAK OCCURRED.
  END;
  *****
  REAL PROCEDURE INPCONV(X); REAL X;
  *****
  BEGIN REAL SEQ;
  REAL STREAM PROCEDURE INCONV(X);
  BEGIN SI:=X; DI:=LOC INCONV; DS:=8UCT; END;
  IF (SEQ:=ABS(INCONV(X))) GTR 99999999 THEN SEQ:=99999999;
  INPCONV:=SEQ;
  END PROCEDURE INPCONV;
  *****
  STREAM PROCEDURE OUTCONV(A,N); VALUE N;
  BEGIN SI:=LOC N; DI:=A; DS:=8DEC; END;
  *****
  STREAM PROCEDURE MOVE(N,A,B); VALUE N;
  BEGIN SI:=A; DI:=B; DS:=N WDS; END;

```

```

00010000
00010100
00010110
00010111
00010112
00010113
00010114
00010115
00010116
00010117
00010118
00010119
00010200
00010300
00010400
00010500
00010600
00010700
00010800
00010810
00010900
00011000
00011010
00011100
00011200
00011300
00011400
00011500
00011600
00011700
00011800
00011900
00012000
00012100
00012200
00012300
00012400
00012500
00012600
00012700
00012800
00012900
00013000
00013100
00013200
00013300
00013400
00013410
00013420
00013430
00013440
00013450
00013460
00013500
00013600
00013700
00013800
00013900
00014000
00014100

```

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

```

%*****00014200
STREAM PROCEDURE INFORMUSER(A,NMBR,LAST); VALUE NMBR, LAST; 00014300
%*****00014400
1 BEGIN LOCAL SV; 00014500
2 DI:=A; SV:=DI; SI:=LOC NMBR; DS:=8DEC; DI:=SV; DS:=7FILL; 00014600
3 SI:=SV; DI:=SV; 8(IF SC="" THEN SI:=SI+1 ELSE DS:=CHR); 00014700
4 DS:=36LIT" RECORDS MERGED (LAST RECORD MERGED="; 00014800
5 SV:=DI; SI:=LOC LAST; DS:=8DEC; DI:=SV; DS:=7FILL; 00014900
6 SI:=SV; DI:=SV; 8(IF SC="" THEN SI:=SI+1 ELSE DS:=CHR); 00014910
7 DS:=16LIT" "; 00014920
8 END STREAM PROCEDURE INFORMUSER; 00014930
%*****00015000
9 STREAM PROCEDURE MAKERR(ERR,N1,N2); VALUE N1,N2; 00015100
%*****00015200
10 BEGIN LOCAL SV; 00015300
11 DI:=ERR; DS:=15LIT"SEQUENCE ERROR:"; SV:=DI; 00015400
12 SI:=LOC N1; DS:=8 DEC; DI:=SV; DS:=7FILL; 00015500
13 DI:=SV; DI:=DI+8; DS:=2LIT" -"; 00015600
14 SV:=DI; SI:=LOC N2; DS:=8DEC; DI:=SV; DS:=7FILL; 00015700
15 END STREAM PROCEDURE MAKERR; 00015800
%*****00015900
16 PROCEDURE ERROR(N1,N2); VALUE N1,N2; REAL N1,N2; 00016000
%*****00016100
17 BEGIN 00016200
18 MAKERR(ERR,N1,N2); TWXOUT(ERR[0],33,1); 00016300
19 END PROCEDURE ERROR; 00016400
%*****00016500
20 A[0]:=0; DISKWAIT(1,A,30,COMMON); % GET ESP RECORD 00016600
21 USER := A[2]; 00016700
22 OUTCONV(LINE,A[1],[40:8]); % DECIMAL LINE NUMBER 00016800
23 IF (MAKTAB:=A[9].[6:12]="15") THEN % TAB FILES REQUIRED 00016810
24 FILL NEWTAB WITH " "&"1"[6:36:12]&LINE[18:30:18],USER; 00016900
25 FILL MERGEIL WITH A[3], A[4]; 00017000
26 FILL OUTFIL WITH A[9], USER; % OUTPUT FILE NAME 00017100
27 FILL SOURCEFIL WITH A[10],USER; 00017200
28 IF RESEQ := A[5] NEQ 0 THEN 00017300
29 BEGIN 00017400
30 STRRESEQ:=A[5]; % LOWER BOUND FOR RESEQUENCE 00017500
31 ENDRSEQ :=A[6]; % UPPER BOUND FOR RESEQUENCE 00017600
32 INCR :=A[8]; % RESEQUENCE INCREMENT 00017700
33 BASE :=A[7]-INCR; % RESEQUENCE BASE; 00017800
34 END; 00017900
35 IF DELTOG := A[11] NEQ 0 THEN % DELETING PART OF SOURCEFILE 00018000
36 BEGIN 00018100
37 STRDELETE := A[11]; 00018200
38 ENDELETE := A[12]; 00018300
39 SUPPRESS := TRUE; % DONT PRINT NUMBER MERGED 00018400
40 END; 00018500
41 NPARAMS := A[1].[27:6]; % PARAMETER COUNT 00018600
42 SFLG := A[1].[3:1]=0; % ON, IF TYPE DATA FILE; 00018700
43 REVTOG := A[1].[5:1]=1; % REVERSE MERGE 00018750
44 SUPPRESS := (A[1].[8:1]=1) OR SUPPRESS; % SUPPRESS MESSAGES 00018800
45 PREVSEQ := NCT := -1; 00018900
46 LREC := 0; 00019000
47 EOFMARK :=100000000; 00019100
48 READ(SOURCEFIL,10,A[*])(EMPTY); 00019200
49 SOURCESEQ := INPCONV(A[9]); GO TO FULL; 00019300
50 EMPTY: SOURCESEQ := EOFMARK; 00019400
51 FULL: 00019500
52 IF SEQCHK := NPARAMS GTR 0 THEN % PARAMETERS SPECIFIED 00019600

```

```

BEGIN
MOVE(9,A[21],PARAMS);
READ SEEK(MERGFIL[0]); EOFPTR:=HDR(MERGFIL,7); % EOF POINTER
REC:=0; NPARAMS:=NPARAMS-1; MERGESEQ:=PC:=-1; LREC:=1;
NEXTPARAM:
LREC := LREC -1;
LRANGE:=HRANGE:=PARAMS[PC:=PC+1];
IF LRANGE=100000000 THEN % "END"
BEGIN
READ(MERGFIL[EOFPTR],14,B[*])[EOFMERG]; READ(MERGFIL);
GO TO HAVEMERG;
END;
IF PC LSS NPARAMS THEN % MORE PARAMETERS AVAILABLE
IF PARAMS[PC+1].[1:1]=1 THEN % RANGE SPECIFIED
HRANGE:=PARAMS[PC:=PC+1].[21:27];
IF NOT SFLG THEN % RECORD LOCATION SPECIFIED
BEGIN
READ SEEK(MERGFIL[LREC:=LRANGE-1]);
GO TO READMERG; % SKIP BINARY SEARCH
END;
IF MERGESEQ = LRANGE THEN % ALREADY HAVE THIS RECORD
BEGIN LREC:=LREC + 1; GO TO HAVEMERG; END;
IF (EOFPTR-LREC) LSS 30 THEN GO TO READMERG; % SKIP BINARY SEARCH
REC:=LREC+(EOFPTR-LREC) DIV 2; % START IN THE CENTER OF RANGE
GO TO TESTREAD;
DO BEGIN
IF LRANGE LSS MERGESEQ THEN % MOVE BACK, NOT LSS 0
REC:=(REC:=REC-ADJ)*REAL(REC.[1:1]=0) ELSE % MOVE UP
IF REC:=(LREC:=REC)+ADJ GTR EOFPTR THEN REC:=EOFPTR;
TESTREAD:
READ(MERGFIL[REC],14,B[*]);
MERGESEQ:=INPCNV(B[9]);
IF MERGESEQ=LRANGE THEN % MATCHING RECORD
BEGIN LREC:=REC; READ(MERGFIL); GO TO HAVEMERG; END;
ADJ:=(REC-LREC) DIV 2;
END UNTIL ADJ LEQ 1;
READ SEEK(MERGFIL[LREC]); % NOT FOUND IN BINARY SEARCH, GO BACK
END; % IF PARAMETERS SPECIFIED
%.....
READMERG:
%.....
READ(MERGFIL,10,B[*])[EOFMERG];
LREC := LREC + 1;
MERGESEQ := IF SFLG THEN INPCNV(B[9]) ELSE LREC;
IF SEQCHK THEN
BEGIN
IF MERGESEQ LSS LRANGE THEN GO TO READMERG; % TOO LOW
IF MERGESEQ GTR HRANGE THEN % TOO HIGH
IF SEQCHK:=PC LSS NPARAMS THEN GO NEXTPARAM ELSE GO EOFMERG;
END; % IF SEQCHK
%.....
HAVEMERG:
%.....
OUTSEQ := MERGESEQ;
NMG := NMG + 1;
IF RESEQ THEN % RESEQUENCE THE MERGE FILE
IF OK OR MERGESEQ GEQ STRRESEQ THEN
IF RESEQ:=OK:=MERGESEQ LEQ ENDRESEQ THEN % RECORDS ARE IN RANGE
BEGIN
OUTSEQ := BASE := BASE + INCR; % CALCULATE NEW SEQ. NUMBER

```

```

00019700
00019800
00019900
00020000
00020100
00020200
00020300
00020400
00020500
00020600
00020700
00020800
00020900
00021000
00021100
00021200
00021300
00021400
00021500
00021600
00021700
00021800
00021900
00022000
00022100
00022200
00022300
00022400
00022500
00022600
00022700
00022800
00022900
00023000
00023100
00023200
00023300
00023400
00023500
00023600
00023700
00023800
00023900
00024000
00024100
00024200
00024300
00024400
00024500
00024600
00024700
00024800
00024900
00025000
00025100
00025200
00025300
00025400
00025500
00025600

```

	OUTCONV(B[9],OUTSEQ); % MOVE NUMBER TO MERGEFILE RECORD	00025700
	END;	00025800
	IF OUTSEQ LEQ PREVSEQ THEN % OUT OF SEQUENCE	00025900
1	BEGIN	00026000
2	IF NOT BRAAK THEN ERROR(PREVSEQ,OUTSEQ); % TELL THE USER	00026100
3	OUTSEQ := PREVSEQ + 2; % ADJUST SEQUENCE NUMBER	00026200
4	OUTCONV(B[9],OUTSEQ);	00026300
5	END;	00026400
6	PREVSEQ := OUTSEQ;	00026500
7	%.....	00026600
8	MERGFILES:	00026700
9	%.....	00026800
10	IF DELTOG THEN IF SOURCESEQ GEQ STRIDELETE THEN % DELETE THIS SECTION	00026900
11	BEGIN	00027000
12	DELTOG := FALSE;	00027100
13	WHILE SOURCESEQ LEQ ENDELETE DO	00027200
14	BEGIN	00027300
15	READ(SOURCEFIL,10,A[*])(EOFSOURCE);	00027400
16	SOURCESEQ := INPCONV(A[9]);	00027500
17	END WHILE STATEMENT;	00027600
18	END DELETING SOURCEFILE SECTION;	00027700
19	IF MAKTAB THEN	00027750
20	IF NPTR := NPTR + 1 GTR 29 THEN % SEGMENT IS FILLED	00027800
21	BEGIN	00027900
22	WRITE(NEWTAB,30,N[*]);	00028000
23	NPTR:=0;	00028100
24	END;	00028200
25	IF SOURCESEQ LSS OUTSEQ THEN % WRITE SOURCE FIRST	00028300
26	BEGIN	00028400
27	WRITE(OUTFIL,10,A[*]);	00028500
28	IF MAKTAB THEN	00028550
29	N[NPTR]:=0 & (NCT:=NCT+1)[4:32:16] & SOURCESEQ[21:21:27];	00028600
30	READ(SOURCEFIL,10,A[*])(EOFSOURCE);	00028700
31	SOURCESEQ:=IF SFLG THEN INPCONV(A[9]) ELSE LREC;	00028800
32	GO TO MERGFILES;	00028900
33	END;	00029000
34	IF SOURCESEQ GTR OUTSEQ THEN % WRITE MERGE RECORD FIRST	00029100
35	BEGIN	00029200
36	WRITE(OUTFIL,10,B[*]); LASTMERGE:=OUTSEQ;	00029300
37	IF MAKTAB THEN	00029350
38	N[NPTR]:=0&(NCT:=NCT+1)[4:32:16]&OUTSEQ[21:21:27];	00029400
39	GO TO READMERG;	00029500
40	END;	00029600
41	% OUTSEQ = SOURCESEQ	00029700
42	MERGREAD:=TRUE; % IN CASE OF EOF BRANCH	00029800
43	IF REVTOG THEN	00029900
44	BEGIN	00029910
45	WRITE(OUTFIL,10,BI[*]); LASTMERGE:=OUTSEQ;	00029920
46	END	00029930
47	ELSE	00030000
48	BEGIN	00030010
49	WRITE(OUTFIL,10,A[*]); NMG:=NMG+1;	00030020
50	END;	00030030
51	IF MAKTAB THEN	00030050
52	N[NPTR]:=0&(NCT:=NCT+1)[4:32:16]&OUTSEQ[21:21:27];	00030100
53	READ(SOURCEFIL,10,A[*])(EOFSOURCE);	00030200
54	SOURCESEQ := INPCONV(A[9]);	00030300
55	MERGREAD:=FALSE;	00030400
56	GO TO READMERG;	00030500
57	%.....	00030600

1	EOFSOURCE:	00030700
2	%.....	00030800
3	IF OUTSEQ=EOFMARK THEN GO TO EXIT; % FINISHED	00030900
4	SOURCESEQ:=EOFMARK;	00031000
5	IF MERGREAD THEN % EOFSOURCE AND SOURCE SEQ. = MERGE SEQ.	00031100
6	BEGIN MERGREAD:=FALSE; GO TO READMERG; END	00031200
7	ELSE GO TO MERGFILES;	00031300
8	%.....	00031400
9	EOFMERG:	00031500
10	%.....	00031600
11	IF SOURCESEQ=EOFMARK THEN GO TO EXIT; % FINISHED	00031700
12	OUTSEQ:=MERGESEQ:=EOFMARK;	00031800
13	GO TO MERGFILES;	00031900
14	%....	00032000
15	EXIT:	00032100
16	%....	00032200
17	IF MAKTAB THEN	00032300
18	BEGIN	00032400
19	IF NPTR:=NPTR+1 GTR 29 THEN	00032500
20	BEGIN	00032600
21	WRITE(NEWTAB,30,N[*]);	00032700
22	NPTR:=0;	00032800
23	END;	00032900
24	NINPTR:=EOFMARK;	00033000
25	WRITE(NEWTAB,30,N[*]);	00033100
26	READ(NEWTAB[0],30,N[*]);	00033200
27	N[0]:=NCT; % EOF POINTER	00033300
28	WRITE(NEWTAB[0],30,N[*]);	00033400
29	LOCK(NEWTAB,*);	00033500
30	END; % IF MAKTAB	00033600
31	CLOSE(SOURCEFIL); CLOSE(MERGFIL); LOCK(OUTFIL,*);	00033700
32	IF NOT SUPPRESS THEN	00033800
33	BEGIN	00033900
34	INFORMUSER(A,NMG, LASTMERGE);	00034000
35	TWOOUT(A[0],52,1);	00034100
36	END;	00034200
37	COMMON := NMG;	00034300
38	END PROGRAM.	00034400
39	END;END. LAST CARD ON OCRDING TAPE	99999999

LABEL 00000000PRINTER00175099CC EX OBJECT/READ;FILE SOURCEFILE=SYMBOL/MERG;END+

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.