DX980

USER-OPERATOR GUIDE

AUSTIN DSD COMPUTER CENTER

JUNE, 1977

TABLE OF CONTENTS

USING THE DX980 OPERATING SYSTEM
USER/OPERATOR GUIDE

The JCL for all utilities in this manual follow each utility. The JCL is included so that the user can modify the utility's resources to accommodate his particular requirements.

All operator responses are underlined throughout this supplement. It assumes the user/operator is using the ASR. If using a CRT, the user is to key in only the underlined portions of each command.

↵ = carriage return.

USING A CRT

ITS

The Interactive Terminal Subsystem (ITS) provides simplified concurrent communication between the DX980 system and a CRT or terminal.

LOGON

When communication is to be established between ITS and a terminal, the terminal user must enter a LOGON command to gain access to ITS facilities. The format of the LOGON command is as follows:

LOGON <USERID> <ACCTNO>

- USERID is an alpha numeric string, limited to 6 characters, the first of which must be alphabetic.

- ACCTNO is a positive integer that must be less than 32,768.

LOGON example: LOGON ANAME 1.↵

If the last CRT user did not LOGOFF, this gives an error message (invalid command). If so, the user should LOGOFF (see LOGOFF command), then LOGON.

Any other special instructions for CRT users will be noted with the JOB being run.

LOGOFF

The LOGOFF command returns the terminal to the inactive state.

LOGOFF example: LOGOFF.↵

ASMBLR (ASSEMBLE)

The general assembler, SAPG, translates 980 symbolic assembly language into object language acceptable to the 980 computer. During pass 1, a symbol table is generated as the source program is read. Pass 2 generates the object output and program listing using both the source program and the generated symbol table. For more details, see: Model 980 Computer Assembly Language Programmer's Reference Manual, No. 943013-9701.

To use the utility ASMBLR:

EXAMPLE: Source from disc file, object to card punch, listing to line printer:

Hit <ESC> for //.

KEY IN:
```
//JOB name SYSTEM.
//RUN ASMBLR FSRC=(userid,filename) DOBJ=CP1.
```

EXAMPLE: Source from cards, no object, listing to mag tape, (using mag tape drive 1; with write ring on mag tape).

KEY IN:
```
//JOB name SYSTEM.
//RUN ASMBLR DIN=CR1 DOBJ=DUMMY DLST=MT1.
```

ASMBLR (980 ASSEMBLER) JCL

```
/REPLACE ASMBLR     .ASSEMBLE.
/EXEC OBJ=(1,SYSTEM,ASMBLR) MEM=(300,5000,1000) PRTY=(1,15);
/          TIME=-1 TIME:=TIM    MEM:=MEM        PRTY:PRI
/ASSIGN  0 DUMMY DEVICE:=DMSG SHARE                              .SYSTEM MESSAGE
/ASSIGN  4 DUMMY DEVICE:=DCON SHARE:=SCON                        .CONTROL/MESSAGE
/ASSIGN  5 DISC1 DEVICE:=DSRC  FILE:=FSRC BUFFERS=1              .SOURCE INPUT
/ASSIGN  6 SYSOUT DEVICE:=DLST FILE:=FLST BUFFERS=1              .SOURCE LIST/ERROR
/                 SHARE:=SLST  MAXTRACK=100
/ASSIGN  7 DISC1 DEVICE:=DOBJ  FILE:=FOBJ  REPLACE:=ROBJ;
/                ACCESS:=COBJ   NEW:=NOBJ ALLOCATE:=LOBJ         .OBJECT OUTPUT
/                ACCESS=(ANY,ANY,ANY,ANY) LINKSEQ BUFFERS=1      .
/                ALLOCATE=(1,0,64,10) FILE=(USER01,ASMOUT)
/ASSIGN 16 DISC1 FILE=(TEMP,SCRL) NEW BUFFERS=1 LINKSEQ;
/          ACCESS=(ANY,ANY,ANY,ANY) ALLOCATE=(5,300,256,50)      .SOURCE SCRATCH
/END
```

ASMCND (ASSEMBLE WITH CONCORDANCE)

This utility is the same as ASMBLR except that a concordance is also generated.

EXAMPLE: Source from disc file, object to card punch, and assembly listing to line printer.

Hit <ESC> for //.

KEY IN:

```
//JOB name SYSTEM.
//RUN ASMCND FSRC=(userid,filename) DOBJ=CP1.
```

EXAMPLE: Source from cards, object to card punch, and assembly listing to line printer.

KEY IN:

```
//JOB name SYSTEM.
//RUN ASMCND DIN=CR1 DOBJ=CP1.
```

ASMCND (980 ASSEMBLER WITH CONCORDANCE) JCL

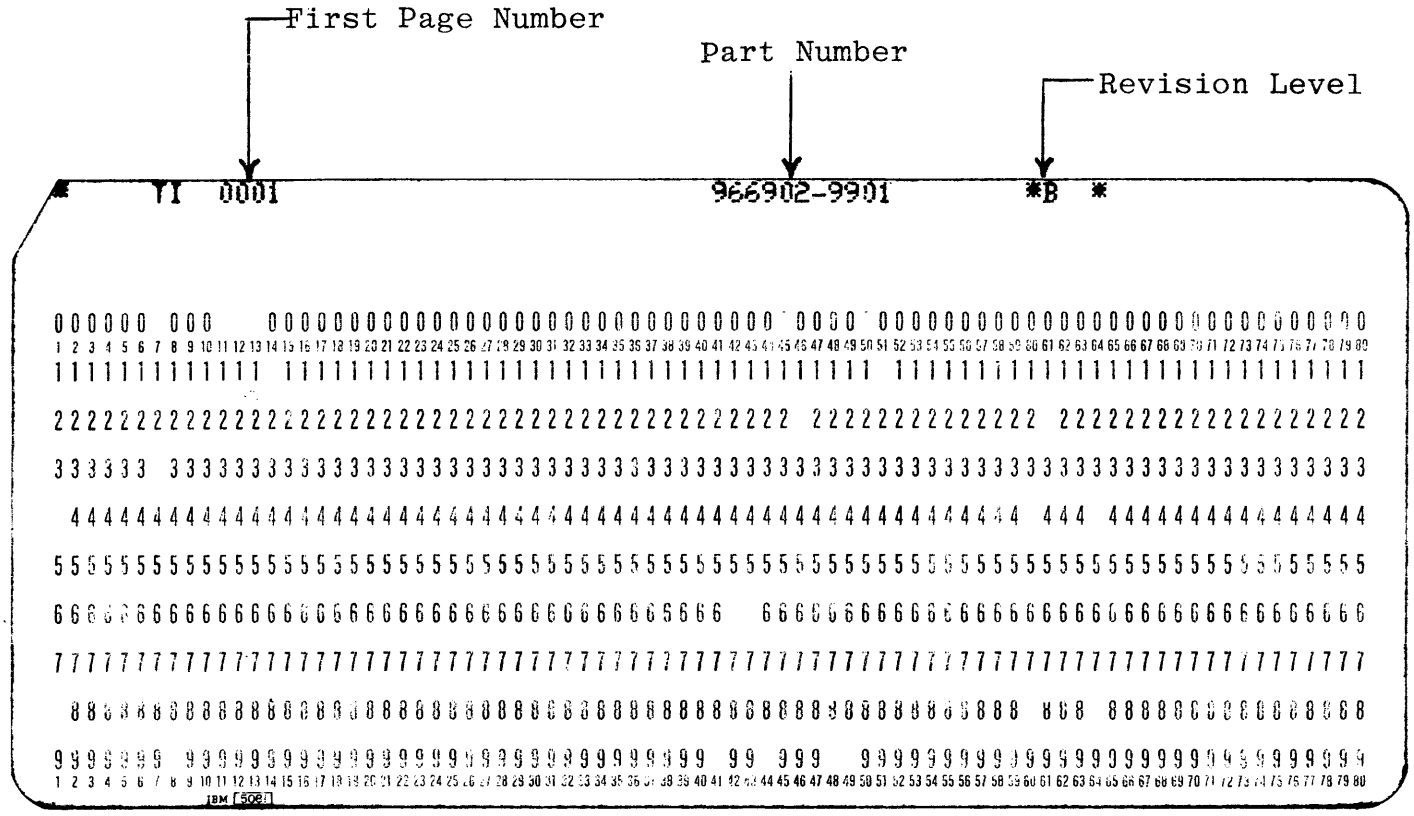
```
/REPLACE ASMCND                                                    .ASSEMBLE.
/EXEC OBJ=(1,SYSTEM,ASMBLR) MEM=(300,5000,1000) PRTY=(1,15);
/          TIME=-1 TIME:=TIM    MEM:=MEM        PRTY:=PRI
/ASSIGN  0 DUMMY                                                   .SYSTEM MESSAGE
/ASSIGN  4 DUMMY DEVICE:=DCON SHARE:=SCON                          .CONTROL/MESSAGE
/ASSIGN  5 DISC1 DEVICE:=DSRC  FILE:=FSRC BUFFERS=1                .SOURCE INPUT
/ASSIGN  6 SYSOUT DEVICE:=DLST FILE:=FLST BUFFERS=1                .SOURCE LIST/ERROR
/                 SHARE:=SLST  MAXTRACK=100
/ASSIGN  7 DISC1 DEVICE:=DOBJ  FILE:=FOBJ  REPLACE:=ROBJ;
/                 ACCESS:=COBJ   NEW:=NOBJ ALLOCATE:=LOBJ          .OBJECT OUTPUT
/                 ACCESS=(ANY,ANY,ANY,ANY) LINKSEQ BUFFERS=1       .
/                 ALLOCATE=(1,0,64,10) FILE=(USER01,ASMOUT)
/ASSIGN 16 DISC1 FILE=(TEMP,SCRL) NEW BUFFERS=1 LINKSEQ;
/          ACCESS=(ANY,ANY,ANY,ANY) ALLOCATE=(5,300,256,50)        .SOURCE SCRATCH
/EXEC OBJ=(1,SYSTEM,CONCRD) MEM=(3,15000,1000) PRTY=(1,15);
/          TIME=-1  PRTY:=PRI
/ASSIGN 16 DISC1  FILE=(TEMP,SCRL)          BUFFERS=1.
/ASSIGN  6 SYSOUT DEVICE:=DLST              BUFFERS=1.
/END.
```

If an error is detected in Pass 1, the message appears before the listing. If the error is detected in Pass 2, the message is printed adjacent to the source line in question. A total number of errors encountered in the assembly is given at the end of the listing.

SAPG ERROR MESSAGES

MESSAGE NUMBER	MESSAGE	MEANING (AND CORRECTIVE ACTION)
1	FIELD SZ	Address beyond reach (use @ for extended format)
2	UNDF OP	Undefined operation code (check list of valid op codes)
3	LONG SYM	Symbol > 6 characters
4	MDF O/F	OPD or FRM multiply defined (rename label)
5	FRM > 16	FRM fields contain more than 16 bits
6	CAD > 10	Address expression > 10 elements
7	UNDF SYM	Symbol not defined (label probably omitted)
8	MDF SYM	Symbol multiply defined (rename labels)
9	RELOC	A relocation error (use only relocatable label in arithmetic expression, or ORG statement can use only one relocatable label)
10	SYM OVF	Too many symbols have been defined (cut out symbols or divide program)
11	BAD NUM	Numeric element not valid (properly define item in label or address field)
12	IMP R/D	A REF or DEF symbol has been used improperly (REF symbol defined inside and outside the program; DEF symbol not defined in the program)
13	X RF USE	A REF symbol has appeared invalidly in an unrelocatable expression
14	IXB ERR	Address mode error (improper use of IXB field)
15	OPD ERR	No such OPD format number
16	ADR MODE	Illegal addressing mode (improperly written address)

EXAMPLE OF TI 980 HEADER CARD



If used, first record of source.

CPYTPE (COPY TAPE TO TAPE)

This program copies tape to tape and verifies. It indicates the number of files copied and verified on the ASR. Default is from MT1 to MT2.

This example will copy and verify to double EOF.

To use: Hit <ESC> for //.

KEY IN:

```
//JOB name SYSTEM.
//RUN CPYTPE.
```

Answer to questions on ASR.

```
User Mode No. = 134 UID-System Name=SC
      COPY/VERIFY MAG TAPE UTILITY

Enter 1 to copy & verify; 2 to verify only  1
```

Enter type of copy

```
EOM   -   up to end of medium
2EOF  -   up to double end of file
N     -   up to and including file N, where N is
          a file number 2EOF
```

When terminated, the program indicates on the ASR:

```
PASS ONE COMPLETE:        N files copied
NORMAL TERMINATION:       N files verified
```

(N = number of files on tape)

NOTE: BE SURE WRITE RING IS OFF MASTER TAPE!

CPYTPE (COPY TAPE TO TAPE) JCL

```
/REPLACE CPYTPE.
/EXEC OBJ=(1,SYSTEM,CPYTPE) MEM=(1,>4100,1000) MEM:=MEM;          .
/               PRTY=(1,15)  TIME=-1 OBJ:=OBJ  PRTY:=PRTY        .
/ASSIGN 0 SC    DEVICE:=DERR                                     .ERRORS
/ASSIGN 4 SC    DEVICE:=DCON FILE:=FCON BUFFERS=1                .CONTROL
/ASSIGN 5 MT1   DEVICE:=DIN                                      .INPUT
/ASSIGN 7 MT2   DEVICE:=DOUT                                     .OUTPUT
/END
```

DXCOPY (FILE COPY UTILITY)

The DXCOPY utility copies data from any type of file or device to any other type file or device. DXCOPY also generates keys for key indexed files.

Listing is defaulted to dummy and DOUT and DIN both are defaulted to DISC1. The default for DCON is dummy which defaults DXCOPY to 'SOURCE' as the control parameter. For further information on DXCOPY consult section 8.7 of the DX980 General Purpose Operating System Programmer's Guide (Manual No. 943005-9701)

Hit <ESC> for //.

* KEY IN:

```
//JOB name SYSTEM.
//RUN DXCOPY FIN=(userid,filename) DOUT=DUMMY;
..DLST=SYSOUT DCON=SC.
```

Answer BEEP with: SOURCE NOKEYS (NO PERIOD HERE!)

EXAMPLE: To copy a binary file from card reader to card punch.

```
//JOB name SYSTEM.
//RUN DXCOPY DIN=CR1 DOUT=CP1 DCON=SC.
```

BEEP: RELOBJ (NO PERIOD HERE!)

* FROM CRT USE DCON=TERMIO

DXCOPY (FILE COPY UTILITY) JCL

```
/REPLACE  DXCOPY        . GENERAL PURPOSE COPY.
/EXEC OBJ =(1,SYSTEM,DXCOPY) MEM=(300,4000,1500) PRTY=(1,15);
/           TIME=-1 PRTY:=PRI    MEM:=MEM
/ASSIGN   0 DUMMY DEVICE:=DMSG                                    .ERROR MESSAGES
/ASSIGN   5 DUMMY DEVICE:=DCON FILE:=FCON BUFFERS=1               .CONTROL/MESSAGE
/ASSIGN   6 DUMMY DEVICE:=DLST FILE:=FLST BUFFERS=1               .LISTING
/           MAXTRACK=200
/ASSIGN   7 DISC1 DEVICE:=DOUT   FILE:=FOUT  REPLACE:=ROUT       .OUTPUT
/                 ACCESS:=COUT KEYLEN:=KOUT ALLOCATE:=LOUT        .
/                 RELREC:=REL INDEXED:=IND   LINKSEQ:=LIN         .
/                  LRECL:=GOUT BUFFERS:=BOUT BUFFERS=2            .
/                 ACCESS=(ANY,ANY,ANY,ANY)    KEYLEN=6           .
/                 ALLOCATE=(1,0,128,10)        LRECL=64
/ASSIGN   8 DISC1 DEVICE:=DIN    FILE:=FIN    BUFFERS=1           .INPUT
/END
```

LIST (80/80 LIST)

Default from cards to line printer, ASCII source. Terminates on /*. Hit <ESC> for //.

KEY IN:

```
//JOB name SYSTEM.
//RUN LIST.
```

Example of List using mag tape input:

```
//JOB name SYSTEM.
//RUN LIST DIN=MT1.
```

This would use mag tape drive 1 as input, with output to line printer.

LIST (80/80 LIST) JCL

```
/REPLACE LIST                  .LIST CARD TO SYSOUT.
/EXEC OBJ=(1,SYSTEM,DXCOPY) MEM=(300,6000,2000) PRTY=(1.15);
/         TIME=-1 MEM:=MEM PRTY:=PRI OBJ:=OBJ
/ASSIGN 0 DUMMY DEVICE:=DMSG                                    .ERROR MESSAGES
/ASSIGN 5 DUMMY DEVICE:=DCON                                    .CONTROL/MESSAGE
/ASSIGN 6 SYSOUT DEVICE:=DLST FILE:=FLST BUFFERS=1;             .LISTING
/         ALLOCATE:=LLIST  MAXTRACK=200.
/ASSIGN 7 DUMMY DEVICE:=DOUT FILE:=FOUT REPLACE:=ROUT;          .OUTPUT
/         BUFFERS=2 BUFFERS:=BOUT LINKSEQ:=LIN RELREC:=REL;
/         INDEXED:=IND ACCESS=(ANY,ANY,ANY,ANY);
/         ALLOCATE=(1,0,128,10) ALLOCATE:=LOUT KEYLEN=6;
/         KEYLEN:=KOUT LRECL=64  LRECL:=GOUT
/ASSIGN 8 CR1    DEVICE:=DIN FILE:=FIN  BUFFERS=1               .INPUT
/END
```

LISTINGS ON WIDE PAPER

Whenever a listing is needed on wide paper (132 col.), first switch the log from the Centronics printer by:

Hit the <ESC> key for //.

KEY IN:

```
//JOB name SYSTEM.↵
//RUN LOG2SC.↵
```

Run the utility desired and after the listing is removed from the printer, switch the log back from the ASR to the Centronics printer by:

KEY IN:

```
//RUN LOG2LP.↵
```

NOTE: From CRT, the last step would be to key in:

```
LOGOFF.↵
```

LOG2SC AND LOG2LP JCL

```
/REPLACE LOG2SC.
/EXEC OBJ=(1,SYSTEM,CHGLOG) MEM=(30,200,200) PRIV PRTY=(1,1) TIME=-1.
/ASSIGN 1 SC.
/END.

/REPLACE LOG2LP.
/EXEC OBJ=(1,SYSTEM,CHGLOG) MEM=(30,200,200) PRIV PRTY=(1,1) TIME=-1.
/ASSIGN 1 LP2.
/END.
```

MANDEL (COPY TAPE TO TAPE)

This copies from mag tape to mag tape, copying all files to double End of File. (If there is not a double EOF, the master tape will run off the end of the tape. It must have a double EOF). After copying all files, both tapes rewind, and the copy is verified. When the copy is verified the tapes rewind and unload. A message on the ASR indicates to the operator/user to swap the tapes and hit carriage return on ASR. The tapes will verify a second time, rewind, unload and the Job terminated. The listing off the line printer is returned with both tapes.

On ASR hit <ESC> key for //.

KEY IN:

```
//JOB name SYSTEM.
//RUN MANDEL.
```

NOTE: Swapping the tapes and reverifying is done to ensure that the copy can be read on another drive other than the one it was written on.

NOTE: BE SURE WRITE RING IS OFF MASTER TAPE!

MANDEL (COPY TAPE TO TAPE) JCL

```
/REPLACE MANDEL
/EXEC OBJ=(1,SYSTEM,CPYTYPE) MEM=(1,>4100,1000)
/     PRTY=(1,15) TIME=-1           PRTY:=PRTY.
/ASSIGN 0 SYSOUT DEVICE:=DALL DEVICE:=DERR.
/ASSIGN 4 DISC1 FILE=(SYSTEM,CPYCN1) BUFFERS=1;
                DEVICE:=DALL FILE:=FCON DEVICE:=DCON.          . CONTROL
/ASSIGN 5 MT1   DEVICE:=DIN.
/ASSIGN 7 MT2   DEVICE:=DOUT.
/EXEC OBJ=(1,SYSTEM,SWAP) MEM=(1,164,150);
/     PRTY=(1,15) TIME=-1 PRTY:=PRTY.
/ASSIGN 0 SC    DEVICE:=DALL  DEVICE:=DSWAP                    . SWAP MESSAGE
/EXEC OBJ=(1,SYSTEM,CPYTPE) MEM=(1,>4100,1000);
/     PRTY=(1,15) TIME=-1           PRTY:=PRTY,
/ASSIGN 0 SYSOUT DEVICE:=DALL DEVICE:=DERR                     . ERR LISTING
/ASSIGN 4 DISC1 FILE=(SYSTEM,CPYCN2) BUFFERS=1;
/               DEVICE:=DALL FILE:=FCON DEVICE:=DCON           . CONTROL
/ASSIGN 5 MT1   DEVICE:=DIN.
/ASSIGN 7 MT2   DEVICE:=DOUT.
/END.
```

MT960 (980 TO 960 MAG TAPE CONVERSION)

This is a generalized version of the program "PRP960". It generates a mag tape on the DX980 system which is in the proper format to serve as input to any 960 system.

If DCON is left at dummy in the JCL, the default record length value is 80 (# bytes/record).

If DCON is assigned to SC or TERMIO, the operator/user specifies a record length (valid length is 1-256 bytes).

Normal run from disc file to mag tape.

On ASR hit <ESC> key for //.

KEY IN:

```
//Job name System.↵
//RUN MT960 FIN=(userid,filename).↵
```

Note: If using cards as original source into the DX980 System, the "!" and " ⅂" characters have opposite Hollerith codes in the 980 and 960 computer systems.

MT960 (980 TO 960 MAG TAPE CONVERSION) JCL

```
/REPLACE  MT960     .980 TO  960 MAG TAPE FORMAT CONVERSION.
/EXEC     OBJ=(1,SYSTEM,MT960);
/                 MEM:=MEM            MEM=(30,643,1000);
/                 PRTY:=PR1           PRTY=(1,15);
/                                     TIME=-1
/ASSIGN 1 DUMMY   DEVICE:=DCON                              .OPERATOR'S CONSOLE.
/ASSIGN 4 DISC1   DEVICE:=DIN:
/                 FILE:=FIN           BUFFERS=1             .DATA INPUT.
/ASSIGN 5 MTI     DEVICE:=DOUT                              .MAG TAPE OUTPUT.
/END
```

MULTI (MULTIPLE COPY LISTING PROGRAM)

This produces multiple copies on the line printer (one after another). Each copy begins at the top of the page.

Default is disc file to line printer.

Hit <ESC> for //.

KEY IN:

```
        //JOB name SYSTEM.
*       //RUN MULTI FIN=(userid,filename).
```

A message is indicated on the ASR asking the operator/user "number of copies desired." Respond with a value greater than 0, from 1 to 3.

On CRT KEY IN:

```
        RUN MULTI FIN=(userid,filename) DCON=TERMIO.
```

(If ITS is running)

MULTI (MULTIPLE COPY LISTING) JCL

```
/REPLACE MULTI  .GENERAL PURPOSE MULTIPLE COPY LISTING PROGRAM.
/EXEC     OBJ=(1,SYSTEM,DXCOPY)     TIME=-1;
/               MEM:=MEMDX          MEM=(300,5000,2000);
/               PRTY:=PRIDX         PRTY=(1,15)                  .DXCOPY PROGRAM.
/ASSIGN 0 DUMMY DEVICE:=DXMSG                                    .DXCOPY ERRORS.
/ASSIGN 5 DISC1 DEVICE:=DXCON       FILE=(SYSTEM,DL980C);        ."SOURCE NOKEYS"
/                                   BUFFERS=1                    .COMMAND.
/ASSIGN 6 DUMMY                                                  .DXCOPY LISTINGS.
/ASSIGN 7 DISC1                     NEW INDEXED KEYLEN=2;
/                                   FILE=(TEMP,MULTI);
/               BUFFERS:=BUFF       BUFFERS=2;
/               ACCESS:=ACC         ACCESS=(ANY,ANY,ANY,ANY);
/               ALLOCATE:=ALL       ALLOCATE=(1,0,128,200)       .ORIGINAL STORAGE.
/ASSIGN 8       DISC1               DEVICE:=DIN;
/               FILE:=FIN;
/               BUFFERS:=BUFF       BUFFERS=2                    .ORIGINAL INPUT.
/EXEC         OBJ=(1,SYSTEM,MULTI)  TIME=-1;
/               MEM:=MEM            MEM=(30,400,1000);
/               PRTY:=PRI           PRTY=(1,15)                  .MULTIPLE COPY PROGRAM.
/ASSIGN 1 SC    DEVICE:=DCON                                     .OPERATOR'S CONSOLE.
/ASSIGN 4 DISC1                     FILE=(TEMP,MULTI);
/               BUFFERS:=BUFF       BUFFERS=2                    .BUFFERED INPUT.
/ASSIGN 6 SYSOUT DEVICE:=DLST;
/               BUFFERS:=BUFF       BUFFERS=2;
/               FILE:=FLST;
/               MAXTRACK:=MAXT      MAXTRACK=100                 .LINE PRINTER OUTPUT.
/END
```

OBJSEQ (OBJECT SEQUENCER)

This program sequences object cards. Default is from card reader to card punch. Increments by 1. Increment number cannot be changed. No control card needed, but PART NUMBER MUST BE SUPPLIED BY USER if needed. The part number can be blanks or a 6 digit number. Program terminates on /*.

If a reply to a device (card reader, card punch, or line printer) must be made, the operator must return the ASR (SC) to operator's mode by:

Hold down <CTRL> key and <O> key, then hit <ESC> key for //. See note for ASR messages.

The sequenced deck will be punched as follows:

col 1	64	reproduced as is from original
col 65		blank
col 66	71	part number
col 72	76	blank
col 77	80	sequence numbers with leading zeros

To run this utility:

Hit <ESC> key for //.

KEY IN:

```
//JOB name SYSTEM.
//RUN OBJSEQ.
```

```
 INPUT PART NUMBER (6 DIGITS) OR SPACE: 123456
LOAD DUPLICATE DECK, KEY GO
GO
OBJECT  SEQUENCE - VERIFY COMPLETED
```

** SEE NOTE ON NEXT PAGE

OBJSEQ (EXAMPLE USING A DISC FILE AS INPUT, CARDS AS OUTPUT)

Hit <ESC> key for //.

KEY IN:

```
//JOB name SYSTEM.
//RUN OBJSEQ DIN=DISC1 FIN=(userid,filename).
```

Answer part number questions by entering a 6 digit part number or blank. Read in duplicate cards after punched for verification when indicated by message on ASR.

NOTE: If any replies to a device (card reader, card punch or line printer) have been made during the job, to get the messages, the operator must return the SC to user mode by:

Hold down <CTRL> key and hit U on console keyboard.

OBJSEQ (OBJECT SEQUENCER) JCL

```
/REPLACE OBJSEQ .980 OBJECT CARD SEQUENCING (WITH OPTIONAL PART #).
/EXEC   OBJ=(1,SYSTEM, OBJSEQ)      MEM=(30,1000,1000);
/                                   PRTY=(1,15)  PRTY:=PRI;
/                                   TIME=-1      MEM:=MEM;
/                                   OBJ:=OBJ
/ASSIGN 4  SC   DEVICE:=DMSG                                  .OPERATOR ANSWERS
/ASSIGN 5  CR1  DEVICE:=DIN         FILE:=FIN  BUFFERS=1      .ORIGINAL OBJECT INPUT
/ASSIGN 6  CR1  DEVICE:=DVER        FILE:=FVER BUFFERS=1      .INPUT FOR VERIFICATION
/ASSIGN 7  CP1  DEVICE:=DOUT        FILE:=FOUT BUFFERS=1      .SEQUENCED OUTPUT
/ASSIGN >16 DISC1                   FILE=(TEMP,SYSTEM);
/                                   BUFFERS=1  NEW;
/                                   ACCESS=(ANY,ANY,ANY,ANY);
/                                   LINKSEQ ALLOCATE:=ALL;
/                                   ALLOCATE=(1,0,32,10)      .SCRATCH FOR STORAGE
/
/END
```

SEQNCR (SOURCE SEQUENCER)

This program sequences source records. Default is from cards to cards. Terminates on /*. Control card is supplied by the user. If a reply to a device (card reader, card punch, or line printer) must be made, the operator must return the ASR (SC) to operator's mode by:

Hold down <CTRL> key and <O> key, then hit <ESC> key for //. See note for ASR messages.

This program increments by 10. The beginning number can be changed, but the increment of 10 cannot be changed.

To use the sequencer: Hit <ESC> for //.

KEY IN:

```
//JOB name SYSTEM.
//RUN SEQNCR DMES=SC.
```

Answer questions on ASR.

```
ENTER CONTROL RECORD ON LUN 4, HIT C/R
READY SOURCE, HIT C/R
LOAD DUPLICATE DECK, HIT C/R
* RE-READ DUPLICATE DECK (YE/NA) ? N   (Y if re-verification is wanted)
```

ERRORS:

If any cards are mis-punched, a duplicate card is punched. Replace the mis-punched one with the duplicate. When the deck has all been read in, a message on the ASR asks if the operator/user wants to re-verify. *Answer with yes (Y) or no (N). If yes, reload the duplicate deck in again, if no, JOB terminates.

NOTE: If any replies to a device (card reader, card punch or line printer) have been made during the job, to get the control messages, the operator must:

Hold down <CTRL> key and hit U on console keyboard.

Using SEQNCR to sequence from a disc file:

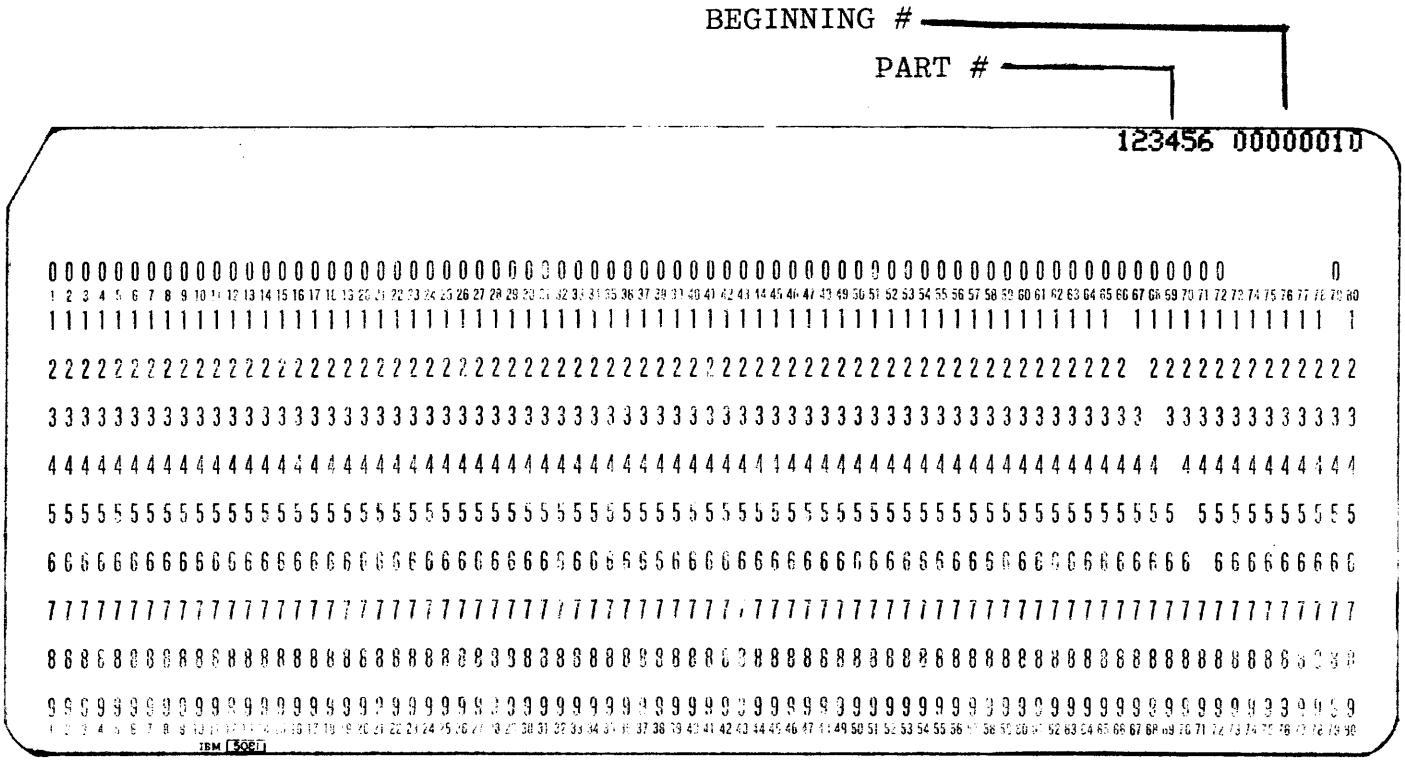
```
//JOB name SYSTEM.
//RUN SEQNCR DMES=SC DSRC=DISC1 FSRC=(userid,filename).
```

This program does not put a /* out after punching the sequenced deck. Because of this, the verify step will not terminate normally. The operator must put a /* card in the card reader to terminate the Job after all of the duplicate deck has read in.

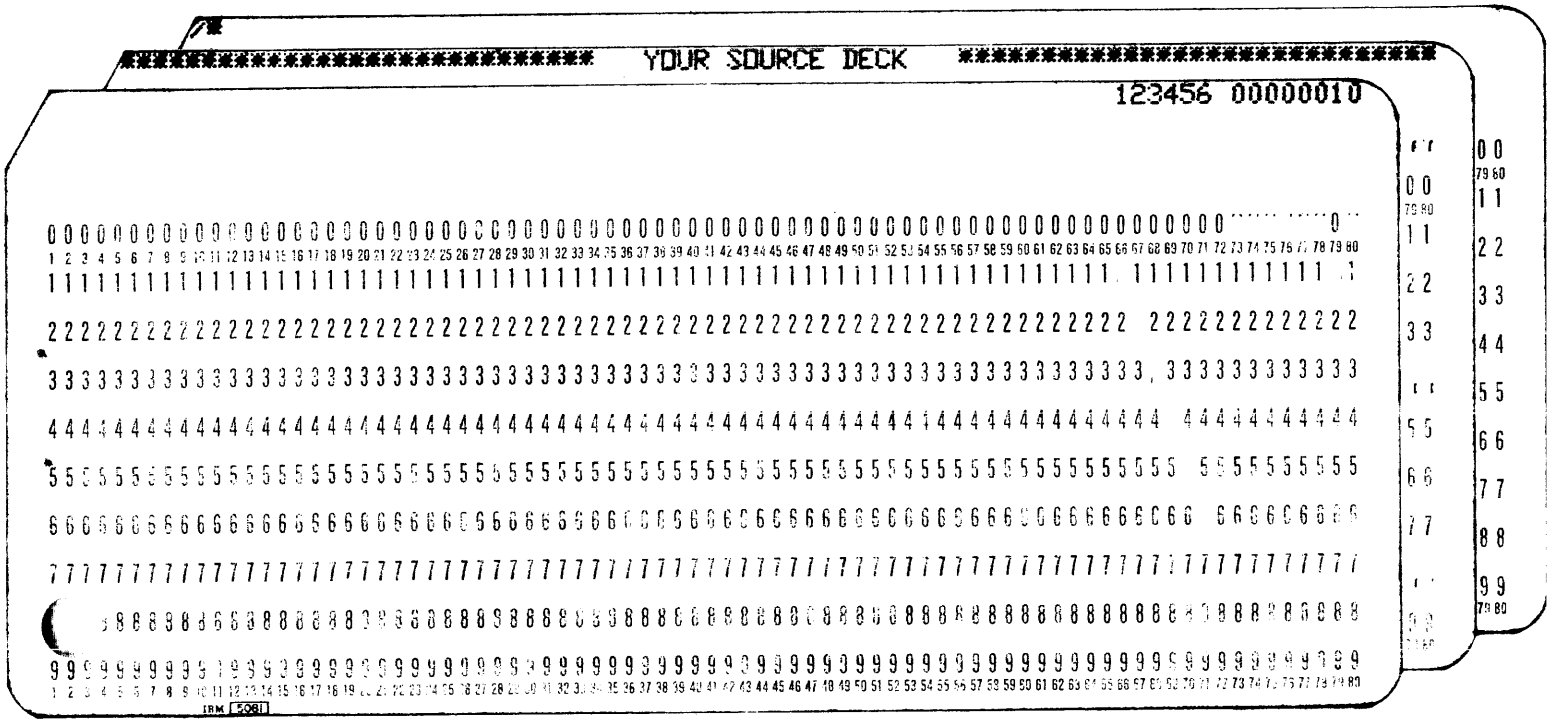
The last record of the file will be punched twice, so the second record should be discarded.

The control card is needed for sequencing from any device.

EXAMPLE OF CONTROL CARD FOR SOURCE SEQUENCER



SEQUENCE EXAMPLE



SEQNCR (SOURCE SEQUENCER) JCL

```
/REPLACE SEQNCR.
/EXEC OBJ=(1,SYSTEM,SEQNCR) MEM=(3,1000,1000)  PRTY=(1,15) TIME=-1  MEM:=MEM;
/                PRTY:=PRI
/ASSIGN  0 DUMMY  DEVICE:=DMES
/ASSIGN  4 CR1    DEVICE:=DCON FILE:=FCON BUFFERS=1
/ASSIGN  5 CR1    DEVICE:=DSRC FILE:=FSRC BUFFERS=1
/ASSIGN  6 CP1    DEVICE:=DCOUT
/ASSIGN  7 CP1    DEVICE:=DOUT
/ASSIGN  8 DISC1  FILE=(TEMP,SCRL) NEW BUFFERS=1 LINKSEQ;
/          ACCESS=(ANY,ANY,ANY,ANY) ALLOCATE=(1,300,256,300)
/ASSIGN  9 CR1    DEVICE:=NSRC
/END
```

VERIFY (RECORD BY RECORD VERIFICATION PROGRAM)

Default run command takes both original and duplicate files from the card reader, lists at most 10 bad records, and assumes input is OBJECT. Gives a listing with a total of records investigated, and satisfactory verification noted when all records match.

Operator to return listing with job.

All records are 80 bytes.

To use hit <ESC> for //.

(FOR OBJECT) Key in: //JOB name SYSTEM.↵
//RUN VERIFY.↵

Answer beep with: DATA NOKEYS↵ (DO NOT PUT A PERIOD HERE!)

To have all errors detected and listed:

(OBJECT) //JOB name SYSTEM.↵
//RUN VERIFY DERR=CR1.↵

Answer beep with: DATA NOKEYS↵ (NO PERIOD HERE!)

FOR SOURCE:

//JOB name SYSTEM.↵
//RUN VERIFY ASCII=CR1.↵

Answer beep with: SOURCE NOKEYS↵ (NO PERIOD HERE!)

This example lists 10 records.

NOTE: DERR (LUNO 1) is defaulted to dummy. This causes the program to automatically terminate after 10 errors. If LUNO 1 is assigned to any other device other than dummy, all errors are listed. DERR may be assigned to any device since no actual I/O is executed with DERR.

ASCII (LUNO 2) is used to indicate whether the records are of Binary or ASCII coding. If left at "Dummy" input is binary. If ASCII is assigned to some device other than DUMMY, input is executed using the "READ ASCII" command. (Default is to DUMMY.)

The DXCOPY utility is first executed to buffer the "original" to a temporary file. This permits verifying inputs from the same device, such as the card reader. If cards are to be used as input, both the original and duplicate decks must end with /*.

VERIFY (RECORD-BY-RECORD VERIFICATION) JCL

```
/REPLACE VERIFY .RECORD-BY-RECORD VERIFICATION PROGRAM.
/EXEC           OBJ=(1,SYSTEM,DXCOPY)      TIME=-1;
/               MEM:=MEMDX                 MEM=(300,5000,2000);
/               PRTY:=PRIDX          PRTY=(1,15) .DXCOPY
/ASSIGN 0 DUMMY DEVICE:=DXMSG        .DXCOPY ERRORS.
/ASSIGN 5 SC    DEVICE:=DCON         .DXCOPY COMMANDS.
/ASSIGN 6 DUMMY                      .DXCOPY LISTINGS.
/ASSIGN 7 DISC1 BUFFERS:=BUFF        BUFFERS=2;
/                                    NEW KEYLEN=2;                .STORAGE
/                                    INDEXED;
/                                    FILE=(TEMP,VERIFY);
/                                    ACCESS=(ANY,ANY,ANY,ANY);
/                                    ALLOCATE=(1,0,128,200)
/ASSIGN 8 CR1   DEVICE:=DOR;
/               FILE:=FORG;
/               BUFFERS:=BUFF        BUFFERS=2                    .ORIGINAL INPUT
/EXEC     OBJ=(1,SYSTEM,VERIFY)      TIME=-1;.VERIFY PROG.
/               PRTY:=PRI            PRTY=(1,15);
/               MEM:=MEM             MEM=(30,2000,2000)
/ASSIGN 1 DUMMY DEVICE:=DERR         .STOP AFTER 10 ERRORS.
/ASSIGN 2 DUMMY DEVICE:=ASCII        .ASCII INPUT IF ASSIGNED.
/ASSIGN 6 SYSOUT DEVICE:=DLST        .ERROR LISTINGS.
/ASSIGN 8 DISC1 BUFFERS:=BUFF        BUFFERS=2;
/                                    FILE=(TEMP,VERIFY);
/                                    ACCESS=(ANY,ANY,ANY,ANY);
/                                    ALLOCATE=(1,0,128,200)
/ASSIGN 9 CR1   DEVICE:=DDUP;
/               FILE:=FDUP;
/               BUFFERS:=BUFF        BUFFERS=2                    .DUPLICATE INPUT
/END
```

A P P E N D E X A

D E V I C E R E P L I E S

CARD READER

If the card reader stops before it is supposed to, there has been either a data error, a read error, or the Job has been rolled.

1. If there is a data error, a message is printed on the ASR (SC). In that case, re-read the last card read and reply to the card reader. (must be from ASR)

2. If there is a read check error, (the read check LT on card reader will be on), re-read the last card and reply to the card reader message indicated on the ASR (SC).

Reply to the card reader by:

Hit <ESC> for //.
KEY IN:

//RE CR1 Y. ↵ (SEE NOTE)

Now it will read more cards.

3. To see if your job has been rolled, get status:

KEY IN:

//ST. ↵
OR: //DS n. ↵ n=JOB number
(SEE APPENDIX B)

NOTE: Reply to terminate the job if needed at this time by:

KEY IN:

//RE CR1 N. ↵

CARD PUNCH

What to do when it stops.

Is it out of cards? If so, put more in, go to clear CP.

Has the Job been rolled? Check STATUS to see if it has been rolled. Hit <ESC> on ASR for //.

KEY IN:

	//ST.	for status of all Jobs in the system.
OR:	//DS n.	for detailed status of a Job if the Job number is known. n=Job number. (SEE APPENDIX B)

Did you have a data error? Look for message on ASR. Clear card punch. Check last card punched before replying to CP.

CLEAR CARD PUNCH

1. HIT <ON LINE> BUTTON
2. HIT <RESET>
3. HIT <CLEAR>
4. REPEAT STEPS 2 AND 3
5. HIT <ON LINE>

NOTE: The <ON LINE> indicator should be the only button lit. If any other buttons are lit, or the <ON LINE> button is blinking, repeat steps 1-5. If there are still any others on, get help.

LINE PRINTER PROBLEMS

To stop a bad listing on the line printer, or if the paper jams:

FIRST STOP THE LINE PRINTER FROM PRINTING.
TURN OFF LINE, WAIT FOR MESSAGE TO COME UP ON ASR (LP1 OFF LINE), TURN ON LINE, THEN HIT ESCAPE KEY (Reply must be from ASR only) for //.

To continue the same listing:

KEY IN:

//BOS SKIP N.↵ (-N for number of lines to back up to; N for number of lines to Skip forward on listing. The position is relative to the line being printed when BOS acts on the received BOS command. Backspacing a number of lines greater than the number already printed from the data set by using the SKIP option is equivalent to the RESTART option; skipping ahead for more lines than the lines remaining to be printed is equivalent to the CANCEL option.

An example of skipping forward 100 lines would be:

//BOS SKIP 100.↵
//RE LP1 YES.↵

An example of backspacing 100 lines would be:

//BOS SKIP -100.↵
//RE LP1 YES.↵

To begin a listing over (page 1) instead of skipping lines,

KEY IN:

```
//BOS RESTART.↵
//RE LP1 YES.↵
```

This command will start at page one of the listing currently on the line printer. If the listing in question has printed the last banner, the job has been deleted in the spooler and the Job will have to be re-run.

To cancel the current listing:

KEY IN:

```
//BOS CANCEL.↵
//RE LP1 YES.↵
```

This will cancel the listing being printed now, and will begin the next listing on the spooler. When this is done, the listing will be deleted and the Job is gone.

APPENDIX B

STATUS

The status command is used to determine the status of Jobs being run in the system. Look for the Job name keyed in at the time a Job was entered, next to the name is the job number.

Status example:

//STATUS. The following is displayed on the ASR or CRT.

```
JOBNAM.JOBNUM.STEP..STATE.....PRI..PRV...LL....EX....UL...TIME USED.
DX980    0      1     RUNNING    0     Y    0000  5C52  6FDA  2190  1934
ITS      1      1     RUNNING    1     Y    7028  9475  A091  200   802
BOS      2      1     RUNNING    2     N    A0B6  A3E4  A6D2  2394  75
RRA      68     1     RUNNING    14    N    A6E5  CBE2  DF6A  29    575
MIKEB    60     1     ROLLED     15    N    A6E2  D5C2  DD92  410   945
JOB NO  60    STEPS  2   - 2  READY
  END OF DISPLAY
```

A detailed status of a particular Job can be done.

Example of the job named MIKEB in the above status example.

//DS 60.

```
JOB NO= 60   IS RUNNING  AT PRIORITY  15
 USER  TASK ID= 0000-RUNNING
END OF DISPLAY
```

A Job can be cancelled by:

//CA 60.

```
  JOB NUMBER  60 CANCELLED
```

This cancelled Job number 60, named MIKEB. A Job cannot be cancelled if it is rolled. It must be running or ready, to be cancelled. Any job can be cancelled from the system console, but if the user wishes to cancel a job from a CRT, the job must have been entered on that CRT.

APPENDIX C

DX980 ERROR CODES

DX980 ERROR CODES

AS OF 01/15/77

1 -OUT OF PARTITION REFERENCE IMPLIED BY PARAMETERS OF AN SVC CALL
2 -JOB EXTENSION AREA TOO SMALL
4 -NO SPACE IN DSCA
5 -ILLEGAL NUMBER OF PARAMETERS IN SVC LIST
6 -I/O ATTEMPTED ON NON-ASSIGNED LUNO
7 -I/O ATTEMPTED WITHOUT OPEN
8 -DUPLICATE OPEN ON SAME LUNO
9 -WAIT CONTROL LIST ERROR FOUND ON USER SUSPEND
10 -PRIORITY ERROR
11 -CPU TIME EXCEEDED
12 -ILLEGAL USER POST
13 -ILLEGAL INSTRUCTION
14 -A NON-EXISTENT SVC WAS ISSUED
15 -USER HAS REQUESTED ACCESS TO A PRIVILEGED SVC
16 -ILLEGAL SVC ARGUMENT -OUTSIDE USER PARTITION
17 -PTR TO SVC ARG LIST OUTSIDE USER PARTITION
18 -INVALID DEVICE I D
19 -NO SPACE IN PCB
20 -NO SYSTEM LUNO = 141
21 -USER FILE DIRECTORY OVERFLOW
22 -MASTER FILE DIRECTORY OVERFLOW
23 -PREVIOUSLY DEFINED USER ID
24 -ILLEGAL USER ID
25 -ATTEMPT TO DELETE SYSTEM DISC MFD
26 -INVALID ABNORMAL JOB TERMINATION CODE
27 -UNDEFINED FILE
28 -UNDEFINED USER ID
29 -ATTEMPT TO REPLACE PREV ASSIGNED FILE
30 -PREVIOUSLY DEFINED FILE
31 -INVALID FILE TYPE OR PHYSICAL RECORD LENGTH > MAX SIZE ALLOWED
32 -INSUFFICIENT TRK SPACE ON DEFINE
33 -INSUFF. CONTIG. TRK SPACE ON DEFINE
34 -EXCEEDED DISC SIZE ON DEFINE
35 -ZERO KEY LENGTH FOR DEFINE
36 -READY JSB FILE BAD
37 -ATTEMPT TO DELETE A SHARED FILE
38 -INVALID FILE DISPOSITION CODE
39 -DEVICE OFFLINE
40 -ATTEMPT TO SHARE UNSHARABLE DEVICE
41 -ATTEMPT TO SHARE BLOCKED DEVICE
42 -ATTEMPT TO ASSIGN EXCL A SHARED PASSED RESOURCE
43 -OPERATOR CANCELLATION
44 -TOO MANY JOB STEPS
45 -INVALID JCB
46 -INVALID INPUT LDT
47 -JOB NO/STEP NO. NOT IN SYSTEM
48 -JOB NAME NOT IN SYSTEM
49 -ATTEMPT TO ILLEGALLY ACCESS FILE
50 -POTENTIAL RESOURCE DEADLOCK DUE TO INCOMPLETE PASSING
51 -INVALID JSB SIZE SPECIFIED
52 -ATTEMPT TO DEASSIGN UNASSIGNED LUNO
53 -TOO MANY JOB STEPS (>15) IN ONE JOB STRING
54 -PARENT JOB ENDED BEFORE JOB STRING STARTED
55 JM-LOAD MODULE TOO BIG FOR SPECIFIED USER SPACE
56 JM-NO JOB INITIATION SYSTEM TASK FOR JOB
57 JM-LOAD MODULE LOAD NO GOOD
58 JM-ATTEMPT TO ASSIGN TO DISC DIRECTLY
59 JM-DEASSIGNMENT OF OPEN DEVICE/FILE

```
 60 SOQM-SYSTEM OUTPUT QUEUE OVERFLOW
 61 SOQM-TOO MANY OUTPUT FILES
 62     -MEMORY PARITY ERROR
 63     -MEMORY PROTECT ERROR - ADDRESSING ERROR
 64     -PRIVILEGED INSTRUCTION VIOLATION
 65     -RESOURCE STACK OVERFLOW
 66     -BYTE RELOCATION ADDR BAD IN LOAD MODULE
 67     -MIP NUMBER BAD FOR LOAD OR LOADR
 69     -LOAD OR LOADR EXTENDS BEYOND USER MEM
 70  BPS-CAN NOT ALLOCATE INPUT DEVICE (REASON-MEMORY OR DEVICE NOT AVAILABLE)
 71  BPS-READ ERROR ON INPUT DEVICE
 72BPSOC-ILLEGAL OR MISSING "JOB" COMMAND
 73BPSOC-ILLEGAL RUN COMMAND
 74  BPS-ILLEGAL DATA COMMAND
 75  BPS-TOO MANY INPUT DATA FILE
 76  BPS-NO OF INPUT DATA FILES UNMATCHED WITH NO OF ASSIGNED INPUT DATA FILES
 77  BPS-DATA COMMAND UNMATCHED WITH INPUT ASSIGNMENTS
 78  BPS-OUTPUT QUEUE ERROR - REINITIALIZE QUEUE TO USE BOS
 80   JM-JOBSTRING/STEP NOT IN ROLL FILE DIRECTORY
 81   JM-NO SPACE AVAILABLE IN ROLL FILE
 82   JM-INSUFFICIENT ROLLABLE MEMORY
 83   JM-ROLL PERFORMED NORMALLY
 84   JM-ROLL FILE CLOBBERED
 85  BPS-DATA ERROR ON LINE PRINTER
 86  BPS-END OF FILE ENCOUNTERED WHILE SKIPPING RECORDS
 87   JM-REQUEST FOR MORE MEMORY THAN IN FREE MEMORY
 88   JM-DEVICE/FILE REQUESTED AT RUN TIME NOT AVAILABLE
 89   JM-INVALID JOB STEP NUMBER IN JSB
 90     -ILLEGAL NUMBER INPUT
 91     -ILLEGAL COMMAND
 92   JM-TOO MANY JOBS IN THE SYSTEM
 93     -LUNO LDT NOT FOUND
 94     -NOT A RR FILE
 95  JCL-USER-ID SPECIFIED FOR PROCEDURE LIBRARY DOES NOT EXIT
 96  JCL-PROCEDURE LIBRARY DOES NOT EXIST UNDER SPECIFIED USER-ID
 97  JCL-USER CANIT GAIN ACCESS TO PROCEDURE LIBRARY BECAUSE OF INTEGRITY CODE
 98  JCL-SPECIFIED PROCEDURE DOES NOT EXIST IN PROCEDURE LIBRARY
 99  JCL-HARDWARE FAILURE WHILE ATTEMPTING READ FROM SPECIFIED PROCEDURE LIB
100   JM-ATTEMPTED TO USE FILE OF RESTRICTED USER ID
101   OC-HARDWARE I/O ERROR IN OP. COMMUNICATIONS
102   OC-INVALID MESSAGE ID
103   OC-INVALID OPERAND IN OP.COMMUNICATIONS
104   OC-INVALID ARGUMENT LIST IN OP.COMMUNICATIONS
105   OC-INVALID JOB NUMBER PASSED IN OP.COMMUNICATIONS
106   OC-ATTEMPT TO OFFLINE SYSTEM DISC OR SYSTEM CONSOLE
107   OC-NO SPACE IN DSCA OR JEA
108   OC-INVALID OP.COMMUNICATIONS COMMAND
109   OC-JOB NUMBER NOT FOUND BY OP.COMMUNICATIONS
110   OC-IN OJCBPR, INVALID SIZE REQUIRED FOR '//JOB' JSB
111   OC-INVALID NUMBER USED FOR SKIP COMMAND TO BATCH OUTPUT SPOOLER
112   OC-UNDEFINED COMMAND GIVEN TO BATCH OUTPUT SPOOLER
113   OC-ATTEMPT TO CANCEL A ROLLED JOB
114   OC-ILLEGAL COMMAND FOR RESERVE OPERATOR COMMUNICATIONS
201   IO-DEVICE NOT READY
202   IO-CONTROLLER ERROR
203   IO-DATA ERROR
204   IO-CONTROLLER BUSY ERROR
205   IO-WRITE PROTECT ERROR
206   IO-EOR ERROR
207   IO-READ-AFTER-WRITE ERROR
208   IO-DEVICE OFFLINE
209   IO-ILLEGAL OP-CODE
210   IO-DEVICE TIMEOUT (DEVICE DID NOT RESPOND)
```

```
233   FM=NO SPACE AVAILABLE ON DISC VOLUME
234   FM=FILE FULL I/O ERROR
235   FM=ATTEMPTED WRITE, LOGICAL RECORD >= PHYSICAL RECORD + OVERHEAD
236   FM=HARDWARE FAILURE ON DISC VOLUME
237   FM=INDEX, REPLACE ATTEMPTED ON NON=EXISTING KEY
238   FM=EXISTING KEY FOUND ON 'WRITE' OP=CODE  ==OPERATION NOT PERFORMED
239   FM=INDEX, WRITE/REPLACE ATTEMPTED ON NON=KEYED RECORD
240   FM=INDEX, REPLACE ATTEMPTED ON KEYED RECORD
241   FM=INDEX, REPLACE ATTEMPTED ON NULL DATA (NON=EXISTENT)
243   FM=INDEX, REL=REC, NO KEY AFTER SEARCH
249   FM=INVALID FILE TYPE (NON=EXISTENT)
250   FM=INSUFFICIENT TRACKS FOR ALLOCATION
251   FM=INSUFFICIENT CONTIGUOUS TRACKS LEFT ON DISC VOLUME
252   FM=ALLOCATION EXCEEDS DISC VOLUME CAPACITY
254   FM=UNABLE TO ALLOCATE BUFFERS BECAUSE OF JOB EXTENSION SIZE
256   FM=INSUFFICIENT NUMBER OF BUFFERS FOR ATTEMPTED OPERATION
257   FM=OPCODE IS EITHER NON=EXISTENT OR ILLEGAL
258   FM=ACCESS VIOLATION, INTEGRITY ERROR
401 CSCN=OVERFLOW OF KEYWORD AREA
402 CSCN=OVERFLOW OF PACKED STRING STORAGE
403 CSCN=R.H.S. OF EXPRESSION OR TERM MISSING
404 CSCN=ILLEGAL EXPRESSION SUBSCRIPT
405 CSCN=MISSING DELIMETER AFTER COMMAND ID
406 CSCN=NUMBER IS LARGER THAN 16 BITS
407 CSCN=OPERAND STARTS WITH ILLEGAL CHARACTER
408 CSCN=ILLEGAL DIGIT IN DECIMAL NUMBER
409 CSCN=MISSING DELIMTER BETWEEN OPERANDS
410 CSCN=MISSING DELIMETER BETWEEN SUBSCRIPTS
411 CSCN=ILLEGAL CHARACTER PRECEEDS COMMAND
412  ITS=RUN COMMAND DOES NOT CONTAIN A LABEL OR AN EXPRESSION
413 CSCN=MISSING EQUAL SIGN IN ASSIGNMENT
414 CSCN=RIGHT HAND SIDE OF ASSIGNMENT MISSING
415 CSCN=MORE THAN ONE = SIGN IN EXPRESSION
416 CSCN=SIZE OF PACKED STRING < 0 CHARACTERS
417 CSCN=UPPER BOUND ON KEYWORD AREA < 1
418 CSCN=NUMBER OF RESERVED LABELS < 0
419 CSCN=STARTING COLUMN FOR SCAN NOT IN RANGE (0:79)
420  JCL=FOR A NEW FILE, USER DIRECTORY NAME DIFFERS FROM CURRENT USER
421  JCL=JSB MUST CONTAIN DEVICE INDEX, NOT THE PDT ADDRESS
422  JCL=DEVICE INDEX MUST BE <= 255
423  JCL=PHYSICAL R.L. < KEY=LENGTH + 14
424  JCL=FILE HAS BAD ACCESS CODE VALUE
425  JCL=BOTH 'DELETE' & 'PASS' SPECIFIED
426  JCL=LOGICAL R.L. > PHYSICAL R.L.
427  JCL=LOGICAL R.L. A MULTIPLE OF 32
428  JCL=DEVICE NOT SPECIFIED OR INCORRECTLY SPECIFIED
429  JCL=USER ID NOT SPECIFIED
430  JCL=FILE NAME NOT SPECIFIED
431  JCL=ILLEGAL FILE NAME SPECIFIED (UFD & MFD ARE NOT VALID)
433  JCL=# PRIORITY LEVELS > 31 OR < 1
434  JCL=JOB STEP PRIORITY > 31 OR < 1
435  JCL=OBJ. VOLUMN ID IS > 20 OR < 1
436  JCL=TCB STACK SIZE < 1 WORD
437  JCL=ILLEGAL COMMAND AFTER 'DELETE'
439  JCL=TIME LIMIT < 1 SECOND
443  JCL=# PRIORITY LEVELS + JOB STEP PRIORITY IS > 32 OR < 1
444  JCL=JOB EXTENSION SIZE <= TCB STACK SIZE + 15 WORDS
445   JCL=INVALID MEMORY PARAMETER
446  JCL=VOL USER ID NOT INITIALIZED
447  JCL=VOLUMN FILE NOT INITIALIZED
448  JCL=VOL USER ID OR FILE NAME > 6 CHARS
449  JCL=VOL PASSWORD > 4 CHARS
450  JCL=WRONG # OF OPERANDS ON THE RIGHT SIDE OF AN EXPRESSION
```

```
451  JCL-OPERAND IS NOT A LABEL OR A SUBSCRIPTED EXPRESSION
452  JCL-OPERAND ON THE RIGHT SIDE IS NOT A LABEL OR A NUMBER
453  JCL-BAD DEVICE NAME
454  JCL-BLOCK SIZE < 1
456  JCL-REDEFINITION OF LUNO IN JOBSTEP
457  JCL-PASSWORD > 4 CHARACTERS
458  JCL-USER ID OR FILE NAME > 6 CHARACTERS
459  JCL-NUMBER BUFFERS < 1
460  JCL-RE-INITIALIZATION OF JSB ITEM
461  JCL-BAD LABEL FOR ACCESS CODE
462  JCL-INITIAL TRACKS < 1
463  JCL-FIRST TRACK ADDRESS < 0
464  JCL-PHYSICAL RECORD LENGTH < 32
465  JCL-PHYSICAL R.L. NOT MULTIPLE OF 32
466  JCL-MAX TRACKS < (INITIAL OR 1)
467  JCL-LOGICAL RECORD LENGTH < 1
468  JCL-LUNO NUMBER NOT IN RANGE 0 TO 254
469  JCL-KEY LENGTH NOT IN RANGE 1 TO 30
470  JCL-OPERAND DOESN'T START WITH LABEL
471  JCL-KEYNAME ON THE LEFT SIDE OF AN EXPRESSION IN NOT DEFINED
475  JCL-OVERFLOW OF KEY-ENTRY TABLE
476  JCL-OVERFLOW OF KEY-REFERENCE TABLE
477  JCL-OVERFLOW OF KEY-CHARS TABLE
478  JCL-DISC ERROR ON LUNO 4
479  JCL-INDEX KEY NAME FOLLOWING 'CREATE' OR 'REPLACE' > 6 CHARACTERS
480  JCL-GREATER 31 ASSIGN CARDS IN THIS JOB STEP
481  JCL-'CREATE' OR 'REPLACE' NOT FOLLOWED BY INDEX KEY NAME
490  JCL-DEV INDEX < 21 OR FILE VOL > 20
491  JCL-PHYSICAL R.L. (CHAR) < (KEYLEN * 2 + 14)
492  JCL-FILE TYPE NOT SPECIFIED WHEN NEEDED (DEFINE, ETC.)
501  JCL-KEYNAME IS NOT IN RESERVED WORD LIST
505  JCL-# OF ENTRIES IN TABLE PASSED TO 'CRLOOK' IS NEGATIVE
510  JCL-SUBSCRIPTS APPEAR ON LHS OF EQUAL SIGN
511  JCL-TRIED TO FETCH NON-EXISTANT RHS SUBSCRIPT
512  JCL-TRIED TO FETCH NON-EXISTANT OPERAND
520  JCL-I/O ERROR ON LUNO = 2.  (ECHO PRINT)
521  JCL-EOF, EOM, OR I/O ERROR ON LUNO = 1.  (JCL IN)
522  JCL-MISSING SLASH IN FIRST COLUMN OF JCL
523  JCL-OVERRIDING KEY WORD ON RUN CARD DOES NOT EXIST FOR THIS PROCEDURE
526  JCL-NUMBER OF KEY-ENTRIES IS NEGATIVE
527  JCL-NUMBER OF KEY-ENTRIES > 19
528  JCL-NUMBER OF KEY-REFERENCES < 0 OR > 19
529  JCL-NUMBER OF KEY-CHARACTERS NOT IN RANGE 0 TO 120
530  JCL-KEY-ENTRY POINTS TO KEY REFERENCE WHICH IS NOT INITIALIZED
531  JCL-KEY-ENTRY FLAG INDICATES THAT NEITHER JCB NOR LDT IS BEING INITIALIZED
532  JCL-LDT NUMBER REFERENCE BY KEY-ENTRY IS NOT IN RANGE 0 TO 30
533  JCL-PRODUCTION NUMBER NOT IN RANGE 3 TO 31
534  JCL-JOB STEP NUMBER NOT IN RANGE 1 TO 15
535  JCL-KEY-CHARS HAS LESS THAN 0 CHARACTERS
536  JCL-CHARACTERS OVERFLOW KEY-CHAR STORAGE
```

APPENDIX D

DX980 DEVICES

DEVICE TABLE

KEY	DEVNAM	ID	TM	ENBLD	EXTREG		EXP	#-SLOT	DEVICE	TYPE
IT	DUMMY	FF	00	Y	0404	DBUS	0000	0002		
DD	DISC1	01	FF	Y		DMAC		0000	DS330	
DD	DISC2	02	FF	N		DMAC		0001	DS31	
MT	MT1	1F	FE	Y		DMAC		0002		
LP	CP1	29	14	Y	4040	DBUS	0000	0006		
CR	CR1	33	FE	Y	1F1F	DBUS	0000	0004		
PR	PTR1	3D	3C	Y	1018	DBUS	0000	0008		HS
PP	PTP1	47	3C	Y	1118	DBUS	0000	0005		HS
LP	LP1	2A	0F	Y		DMAC		0005		
LP	LP2	2B	0F	Y	5050	DBUS	0000	0007		
CS	CS11	65	3C	Y	0505	DBUS	0000	0003		
CS	CS12	66	3C	Y	0505	DBUS	0000	0003		
DT	SC	15	FF	Y	0505	DBUS	0000	0003	733 -	CM
DT	CRT1	16	FF	Y	0606	DBUS	0001	0005	CRT -	CM
DT	CRT2	17	FF	Y	0707	DBUS	0000	0009	CRT -	CM
DT	CRT3	18	FF	Y	0808	DBUS	0000	000A	CRT -	CM
DT	CRT4	19	FF	Y	0909	DBUS	0000	000B	CRT -	CM
DT	KEY1	1A	FF	Y	0B0B	DBUS	0000	000D	733 -	CM
MT	MT2	20	FE	Y		DMAC		0002		
MT	MT3	21	FE	N		DMAC		0002		
DT	CRT5	1B	FF	Y	0C0C	DBUS	0001	0006	CRT -	CM
DT	CRT6	1C	FF	Y	0D0D	DBUS	0001	0007	CRT -	CM
DT	CRT7	1D	FF	Y	0E0E	DBUS	0001	0008	CRT -	CM

SYSTEM DISC = DISC1

LOGGING DEVICE = SC

SYSTEM CONSOLE = SC

APPENDIX E

JOB TERMINATION CODES

JOB TERMINATION CODES

When a job terminates, the system generates a message to the log which gives the termination code for that particular job.

An example of a normal job termination:

```
NO. =30  UID= MIKEB  NAME=MIKEB  SUBMITTED

NO. =30  UID= MIKEB  NAME=MIKEB  STEP=1  TERMINATED AT  0F03
*   TERM CODE=0    TASK TIME= 0  5 21  377 MJAR=4067  MJEA=821  EX=9112

STRING NO. = 30  TERMINATED
```

* A term code other than 0 indicates abnormal job termination. See Appendix C for error codes.

The purpose of this guide is to aid the users of the Austin DSD Computer Center. For information not contained in this guide, reference manuals:

DX980 General Purpose Operating System
System Operating Guide, no. 943004-9701

DX980 General Purpose Operating System
Programmers Guide, no. 943005-9701