IBM Systems Reference Library

IBM 1800 Multiprogramming Executive Operating System
Operating Procedures
Fifth Edition (August 1974)

This is a reprint of GC26-3725-3 incorporating changes released in the following Technical Newsletters:

GN34-0095 (October 1974)
GN34-0188 (February 1974)

This edition applies to version 3, modification 4, of the IBM 1800 Multiprogramming Executive Operating System, and to all subsequent versions and modifications unless otherwise indicated in new editions or technical newsletters. Changes may be made to the specifications in this manual at any time; before using this manual in connection with the operation of IBM systems, consult the latest SRL Newsletter, GN26-800, for the editions that are applicable and current.

A form for readers' comments is provided at the back of this publication. If the form has been removed, send your comments to IBM Corporation, Systems Publications, Department 27T, P. O. Box 1328, Boca Raton, Florida 33432. Comments become the property of IBM.

Requests for copies of IBM publications should be made to your IBM representative or the IBM branch office serving your locality.

©Copyright International Business Machines Corporation 1970, 1971
Preface

This manual describes operating procedures for the IBM 1800 Multiprogramming Executive (MPX) Operating System. It contains procedures for system generation, reload, Cold Start, batch and background processing, maintenance, utility, and debugging operations.

Before using this book, you should be familiar with the basic concepts and terms associated with MPX. They're explained in the MPX System Introduction manual, Order Number GC26-3718. You should also be familiar with the machine units that make up your 1800 system. They're described in the 1800 System Summary, Order Number GA26-5920.

While using this book to operate the 1800, you'll need some other books:

1800 Functional Characteristics, Order Number GA26-5918, which describes how the physical units of the 1800 system work.

1800 Operating Procedures, Order Number GA26-5953, which tells how to operate the physical units.

MPX Error Messages and Recovery Procedures, Order Number GC26-3727, which lists all the error messages MPX prints and tells what conditions they indicate and how to recover from these conditions.

MPX Control Statements, Order Number GX26-1594, which gives the format of each of the Supervisor, DMP, Builder, Macro Assembler, FORTRAN Compiler, Cold Start, and System Loader control statements.

As you read this manual, you might want more detailed information about 1800 system physical units and MPX system programs. These books can be used for reference:

MPX Programmer's Guide, Order Number GC26-3720, which discusses MPX system organization and programming techniques.

MPX Planning for Versions 2 and 3, Order Number GC26-3731, which describes the features of MPX added in Versions 2 and 3 of the system.

MPX Subroutine Library, Order Number GC26-3724, which describes each of the system subroutines.

1130/1800 Assembler Language, Order Number GC26-3778, which tells how to use macro instructions and write programs in assembler language.

1130/1800 Basic FORTRAN IV Language, Order Number GC26-3715, which tells how to write programs in FORTRAN.

Communications Adapter Programming, Order Number GC26-3757, which tells how to write programs to carry out communications with other computers and terminals.

Binary Synchronous Communications — General Information, Order Number GA27-3004, which describes the programming conventions that govern communications between the 1800 and other computers and terminals.

1130/1800 Plotter Subroutines, Order Number GC26-3755, which describes MPX subroutines for controlling the 1627 plotter.

This book consists of eight chapters and four appendixes.
The first chapter, "I/O Device and Console Operations", describes procedures for various operations on system I/O devices and on the 1800 console.

The second chapter, "MPX System Generation," describes the process of defining and establishing the capabilities, contents and organization of your particular MPX system from materials supplied by IBM Program Information Department. You don't need to read this chapter unless you're going to be carrying out a system generation.

The third chapter, "Cold Start," describes the process of starting your batch-processing system (under BOM) or your real-time system (under the Executive) once your system has been generated.

The fourth chapter, "BOM/Executive Reload," summarizes the functions performed during a system reload and details manual operating procedures for performing a reload.

The fifth chapter, "Background-Processing Operations," discusses the procedures for passing control to the Batch-Processing Monitor Supervisor. It also discusses the types of input to the Supervisor.

The sixth chapter, "DMON — System Maintenance Program," describes the operating procedures for the DMON program, which is used to alter and update system programs.

The seventh chapter, "BOM Card Utilities," describes the functions and operating procedures for the various BOM card utility programs punched during system generation.

The eighth chapter, "MPX Debugging Aids," describes the functions and operating procedures for the various system debugging aids.

System Loader assignment cards, decimal and hexadecimal disk addresses, data formats, and an MPX sample program are described in the appendixes.
Contents

I/O DEVICE AND CONSOLE OPERATIONS
I/O Device Operations .................................................. 1
1442 Card Read Punch, Models 6, 7 .................................... 1
1816/1053 Printer ......................................................... 5
1054 Paper Tape Reader .................................................. 6
1055 Paper Tape Punch .................................................. 7
1810 Disk Storage Unit .................................................. 8
1827 Plotter .............................................................. 9
2401 Magnetic Tape Unit ................................................. 10
1443 Printer ............................................................. 11
2311 Disk Storage ....................................................... 13
2790 I/O Devices ....................................................... 14

Console Operations ..................................................... 15
Clear Main Storage ...................................................... 15
Console Program Load (Not Under MPX) ............................... 15
Altering or Displaying the Contents of Main-Storage Locations .................................................. 16
Operations Monitor ...................................................... 16

MPX SYSTEM GENERATION
Introduction ................................................................... 17
Generating Your System .................................................. 17
Materials ........................................................................ 18
Cards ........................................................................... 18
Disks ........................................................................... 18
Cards ........................................................................... 21
Scratch Disks ............................................................... 21
Operating Procedures ..................................................... 21
Phase 1 Operations ....................................................... 22
Phase 2 Operations for Single 1810 Drive System ................. 23
Phase 2 Operations for 2311 or Two or More 1810 Drive Systems .................................................. 29
Phase 3 Operations ....................................................... 44
Phase 4 Operations ....................................................... 45
Phase 5 Operations ....................................................... 46

COLD START .................................................................... 51
Cold Starting a Batch-Processing System ......................... 51
Cold Starting a Real-Time System .................................... 51
Using a Cold Start Coreload ............................................ 52
Cold Start Card Formats ................................................ 52
Cold Start Loader Cards ............................................... 52
Cold Start Name Card .................................................. 52
Operating Procedures .................................................. 55
Cold Start Messages ...................................................... 57

BOM/EXECUTIVE RELOAD
BOM Reload ............................................................... 59
Absolute Loader ........................................................... 59
Batch-Processing Monitor .............................................. 59
Set Clock ................................................................. 60
Set Date ........................................................................ 60
Executive Reload ....................................................... 61

BACKGROUND-PROCESSING OPERATIONS ......................... 63
Loading the Supervisor ............................................... 63
Input to the Supervisor ................................................ 63
Keyboard Input ................................................................ 64
OPTIONAL MATERIAL TAPE DUMP PROGRAM .......................... 110.1
Operating Procedures ................................................. 110.1
Output Format ......................................................... 110.2
Error and Termination Procedures ................................. 110.2
   Termination of the Current Module ............................. 110.2
   I/O Error Retry .................................................. 110.2
   MOPTP Errors Detected .......................................... 110.2
Trap Subroutines ..................................................... 111
CE Coreload Programs (CECLD, CECLX) ......................... 115
   How to Build CE Coreloads ..................................... 115
   How to Use a CE Coreload ....................................... 116
   CE Coreload Functions .......................................... 116
1442 Reader Diagnostic ............................................. 124
   Operating Procedures ........................................... 124
   1442 Diagnostic Error Messages ................................. 125

APPENDIX A. SYSTEM LOADER ASSIGNMENT CARDS ................... 127
APPENDIX B. DECIMAL AND HEXADECIMAL DISK ADDRESSES ......... 131
APPENDIX C. DATA FORMATS .......................................... 133
   Disk System Format (DSF) ....................................... 133
      Program Header Format ....................................... 133
   Disk Data Format (DDF) .......................................... 135
   Card System Format (CDS) ....................................... 135
      Mainline Header Card ......................................... 135
      Data Cards ................................................... 136
      EOP Card ..................................................... 136
      Subroutine Header Card ..................................... 137
      ISS Header Card ............................................. 137
      Sector Break Cards .......................................... 137
   Card Data Format (CDD) .......................................... 138
      Disk Dump for Physical or Mapped 1810s .................... 138
      Disk Dump for 2311s ......................................... 139
      Main Storage Dump ............................................ 140
   Print Data Format (PRD) ......................................... 140

APPENDIX D. MPX SAMPLE PROGRAM ................................ 141
Operating Procedure ................................................. 141

APPENDIX E. SUMMARY OF CONSOLE SENSE, PROGRAM, AND DATA
SWITCH OPERATIONS ................................................ 150.1

GLOSSARY-INDEX ..................................................... 151
Illustrations

Figures

Figure 1. BOM High Core Loader Cards ........................................ 19
Figure 2. Define Number of Drives and Label Disk Cartridges ................. 45
Figure 3. Define Main-Storage and Disk Layout ................................ 46
Figure 4. BOM Build Job .......................................................... 47
Figure 5. Batch-Processing System Cold Start Deck ............................. 47
Figure 6. Assemble, Compile, and Store Executive Subroutines on Disk .... 48
Figure 7. Building the Executive ............................................... 49
Figure 8. Compile and/or Assemble User-Written Programs, and Build .... 50
Figure 9. Cold Start Loader Cards ............................................. 53
Figure 10. Sequence of Input Statements for a System Program Patch ....... 71
Figure 11. Input for a System Program Update ................................ 75
Figure 12. Deck for Updating the Subroutine Library .......................... 76
Figure 13. Job Stream for DMPAN ............................................ 105
Figure 14. DMPAN Control Card Format .................................... 105
Figure 15. Using CECLD ....................................................... 121
Figure 16. Using CECLX ....................................................... 122
Figure 17. Sample Complement Input Card for 1442 Reader Diagnostic .... 124
Figure 18. Disk System Format ................................................ 133
Figure 19. Card Data Format ................................................... 138

Tables

Table A. Communications Adapter Completion and Error Codes .............. 122.2
Table B. 2311 Error Conditions ................................................ 122.3
Table C. 2841/2311 Command Codes ......................................... 122.4
Table D. 2841/2311 Status and Sense Bytes ................................ 122.5
Table 1. IAC/LUN Assignments for MPX System .............................. 128
Table 2. Assignment Card Format .......................................... 129
I/O Device and Console Operations

Operations on an 1800 MPX system require a familiarity with some I/O device and console operations.

I/O Device Operations

Loading, readying, error recovery, and unloading procedures for various I/O devices follow.

1442 CARD READ PUNCH, MODELS 6, 7

Readying and error recovery procedures for the 1442 card read punch follow.

PRE-CONDITIONS
System power ON, CHECK light OFF, CHIP BOX light OFF, stacker not full, covers closed.

READYING PROCEDURE
1. Place cards to be read or punched into the hopper, face down, 9-edge first.
2. Press reader START.
The READY light turns on when the first card is positioned at the read station.

1442 ERRORS AND RECOVERY PROCEDURES

If a 1442 error occurs, the 1442 becomes not ready until the operator has intervened. Unless the stop is caused by a stacker full (no indicator) or chip box indication, the 1442 card path must be cleared before proceeding. The 1442 error indicators and the position of the cards in the feed path should be used to determine which cards must be placed back into the hopper.

As far as the card subroutines are concerned, a retry consists of positioning the cards as indicated in the following paragraphs and reinitiating the read or punch operation when the unit becomes ready. The card subroutines will skip the first card, if necessary, on a read or feed operation.
HOPPER MISFEED

Indicates that card 2 failed to pass properly from the hopper to the read station during the card 1 feed cycle.

Card positions after error:

Error indicator: HOPR

Recovery procedure: Empty hopper, press NPRO to eject card 1, replace card 1 into the deck, return deck to hopper, and ready the 1442.

FEED CHECK (PUNCH STATION)

Indicates that card 1 is improperly positioned in the punch station at the completion of its feed cycle.

Card positions after error:

Error indicator: PUNCH STA

Recovery procedure: Empty hopper, clear 1442 card path. If reading, place card 2 into hopper before card 3, and ready the 1442. If punching, place cards 1 and 2 in hopper before card 3 and ready the 1442.

TRANSPORT

Indicates that card 1 has jammed in the stacker during the feed cycle for card 2.
Card positions after error:

![Diagram showing card positions after error]

Error indicator: TRANS

Recovery procedure: Empty hopper, clear 1442 card path, place cards 2 and 3 into the hopper, and ready the 1442.

FEED CYCLE

Indicates that the 1442 did an uncalled-for feed cycle, resulting in cards 1, 2, and 3 being each one station farther ahead in the 1442 card path than they should be.

Card positions after error:

![Diagram showing card positions after error]

Error indicator: FEED CLU

Recovery procedure: Empty hopper, press NPRO to eject cards 2 and 3, place cards 1, 2, and 3 into the hopper, and ready the 1442.

FEED CHECK (READ STATION)

Indicates that card 1 failed to eject from the read station during its feed cycle.
Card positions after error:

Error indicator: READ STA

Recovery procedure: Empty hopper, clear 1442 card path, place cards 1 and 2 into the hopper, and ready the 1442.

READ REGISTRATION
Indicates incorrect card registration or a difference between the first and second reading of a column.

Card positions after error:

Error indicator: READ REG

Recovery procedure: Empty hopper, press NPRO to eject card 1, place card 1 into the deck in front of card 2, return deck to hopper, and ready the 1442. Repeated failures of this type might indicate a machine malfunction.

PUNCH CHECK
Indicates an error in output punching.
Card positions after error:

![Diagram of card positions]

**Error indicator:** PUNCH

**Recovery procedure:** Empty hopper, check card position and press NPRO to clear 1442 card path. If necessary, correct card 1 to prepunched state. Place corrected card 1, and card 2 into the hopper before card 3 and ready the 1442.

**OVER RUN**

Indicates that a cycle steal request was not honored in time and that data has been lost. Card positions indicate the kind of operation, read or punch.

If the operation was a read, the card positions after the error and the recovery procedures are the same as for Read Registration.

If the operation was a punch, the card positions after the error and the recovery procedures are the same as for Punch Check.

**Error indicator:** OVER RUN

**PARITY**

Indicates incorrect parity detected on data transfer to or from B-register. See "Read Registration" or "Punch Check" for card positions after error and for recovery procedures. The card read punch will be ready.

**Error indicator:** NONE

**STORAGE PROTECT**

Indicates attempt to read a column into a storage protected location. See "Read Registration" for card position after error and operator recovery procedures. The card read punch will be ready.

**Error indicator:** NONE

**1816/1053 PRINTER**

Readying and loading procedures for the 1816/1053 printer follow:

**PRE-CONDITIONS**

System power ON, 1816 motor switch ON.
READYING PROCEDURE

Press CARRIER RETURN.

LOADING PROCEDURE

1. Open the 1816/1053 top cover.

2. Pull the paper pressure rod (the rod with three rubber rollers that leans against the platen) forward. If the paper is to be pin fed, this rod should remain in this position.

3. Lift up on the left and right platen pin feed pressure plates.

4. Set the paper release lever in the forward position. This lever is located on the top right rear corner of the 1816/1053. If the paper is to be pin fed, this lever should remain in this position.

5. Feed the paper in from the rear and guide it under the platen. Make sure that the paper lies over and closes the Forms Check microswitch.

6. Lay the paper back across the top of the 1816/1053 and guide the paper so that the holes line up with the pin feeds.

7. Close the pin feed pressure plates.

8. Looking directly down into the 1816/1053 set the left and right margins. The margin settings can be read on the scale across the front of the unit. Use the TAB key to move the carrier off the left margin.

9. Close the top cover.

1054 PAPER TAPE READER

The readying procedure follows.

PRE-CONDITIONS

System power ON.

PROCEDURE

1. Place the reel of tape on the supply reel on the right side of the reader. The tape feeds under and out towards the front of the unit.

2. Feed the tape under the guide in front of the reel and bring it up over the side and across the unit.

3. Press the TABLE RELEASE key.

4. Slip the tape under the plate and position the tape so that a delete code in the reader is over the read head. If you are starting to read at a point other than at the beginning of the tape, place the first character to be read over the read head.

5. Press down on the table to lock the tape in place.
1055 PAPER TAPE PUNCH

The readying and loading procedures follow.

PRE-CONDITIONS

System power ON.

READYING PROCEDURE

To make a leader (all delete codes):

1. Press and hold the DELETE key.
2. Press and hold the FEED key until a leader of sufficient length has been punched.
3. Release the FEED key.
4. Release the DELETE key.

LOADING PROCEDURE

1. Place a reel of tape in the supply pan so that the tape feeds out towards the punch die (see illustration below).
2. With the punch die facing forward (unit name plate at the front), pivot the tape pressure lever (right side of die) up and to the right.
3. Feed the tape from the supply pan over the first tape guide and under the tape tension lever, and slide the tape in under the punch die, tear guide, and tape pressure lever.
4. If the punch has a takeup reel, guide the tape over the side of the unit, over the outside of the side guide, and back up toward the front of the unit.
5. The tape now makes a half turn toward the outside and comes up and over the end guide.
6. The tape is then brought up and over to the left and wound over the top of the take-up reel.
1055 Paper Tape Punch

1810 DISK STORAGE UNIT

The readying and loading procedures follow.

PRE-CONDITIONS
System power ON, CARTRIDGE UNLOCKED lights ON.

LOADING PROCEDURE
1. Open the front door of the disk unit.
2. Grasp the handle of the access release mechanism of the drive to be loaded (drive 2 on top, 0 in the center, 1 on the bottom) and pull out and down.
3. Pick up the cartridge and, holding the cartridge with the IBM name toward you and on the left, insert the cartridge into the slot.
4. When the cartridge is positioned, raise the access release handle and lock the cartridge into place. If desired, load the other drives on the 1810 disk storage unit.
5. Press START on the desired drive.
The CARTRIDGE UNLOCKED lights will go out when the drives start to turn. When the drives come up to speed (approximately 90 seconds), the indicators showing the drive numbers will light, thus showing that the heads are loaded and the drives are ready.

1627 PLOTTER

The readying and loading procedures follow.

PRE-CONDITION
System power ON.

READYING PROCEDURE
1. If the pen is not in a raised position, turn the pen switch, first DOWN and then UP.
2. With the pen in the UP position, use the drum (X axis) and carriage (Y axis) controls to position the pen for the first plot. The 1627 plotter is now ready to be selected.

If you wish to load a new paper roll, use the following procedure:

LOADING PROCEDURE
1. Turn the 1627 power switch OFF (1627 power on indicator lamp out).
2. Remove the pen assembly, if installed, by loosening the knurled knob at the bottom of the pen holder and lifting the assembly out of the carriage.

CAUTION: Use care when handling the pen assembly; it is manufactured to close tolerances for optimum performance.
3. Rotate the right rear chart spool by hand until the drive key is pointing upward.
4. Hold the new roll of chart paper so that the key slot in the core is pointing upward. Place the roll against the spring-loaded left rear idler spool and force the spool to the left.
5. Lower the paper roll into the paper well and slide the right end onto the drive spool. Make certain the drive key engages the key slot in the core. The paper should feed out from under the roll and over the drum (see illustration).

I/O Device Operations 9
6. Install a paper roll core on the front spool below the drum, in the same manner as with the paper roll.

7. Pull a short length of paper off the roll, slide the end under the carriage rods, under the tear bar, and behind the core, and fasten it to the front side of the core with two or three short pieces of cellophane tape. Wind one or two turns of paper onto the core. Make certain the drum sprockets are properly meshed with the sprocket holes on both sides of the paper.

8. Reinstall the pen assembly in the carriage.

9. Turn the 1627 power switch to ON. The 1627 power on indicator will come on.

Note: The pen is down when the power is off; therefore, the pen assembly should be installed with the carriage over an area outside the "recording area." If the pen does not raise when power is turned on, turn the pen switch to DOWN, then to UP.

2401 MAGNETIC TAPE UNIT

The readying, loading, and unloading procedures follow.

PRE-CONDITIONS

System power ON.

2401 Magnetic Tape Unit

READYING PROCEDURE

1. Press LOAD REWIND. The read head will lower, the capstan drives will engage, the tape will be lowered into the vacuum columns, and the tape will rewind to load point.
2. Press START. The tape READY light will turn on. The tape drive is now ready to be selected.

LOADING PROCEDURE

1. Open the front door of the magnetic tape unit.
2. Unlock the takeup and supply reel hubs (see illustration).
3. Place the reel of tape to be read on the supply reel (left side). The end of the tape should hang down from the right side of the reel. If file protection is desired (read from this tape only), ensure that there is no file protect ring on the back of the tape. The groove around the back of the reel should be empty. If you desire to read or write with this tape, ensure that the file protect ring is in place. The FILE PROTECT light will go out when the unit is ready.
4. Lock the supply reel hub.
5. Place an empty tape reel on the takeup reel (right side).
6. Lock the takeup reel hub.
7. Press the reel release button (lower left of supply reel) and unwind about four feet of tape.
8. Thread the tape according to the illustration and, holding the end of the tape against the center of the takeup reel with one finger, wind the tape onto the takeup reel. The reel release button must be held down or the tape will not move. If this reel of tape has been used before, watch for the load point reflective marker on the outer edge (the side nearest you) of the tape. The load point marker must be to the right (takeup reel side) of the read head before the tape drive can be successfully loaded. If this is a new reel of tape, wind the tape until it is securely held on the takeup reel and attach a load point marker to the front edge of the tape.
9. Close the front door of the magnetic tape unit.

UNLOADING PROCEDURE

1. Press RESET (the unit cannot be unloaded while the unit is ready).
2. Press UNLOAD.

If you want to reload a tape you have just unloaded, manually reposition the load point marker to the right of the read head. If you attempt to load the tape with the load point marker to the left of the read head, the tape will unwind from the takeup reel.

1443 PRINTER

Readying and loading procedures follow:

PRE-CONDITIONS
System power ON, ribbon in place, typebar in place, typebar motor switch ON.

READYING PROCEDURE
Press printer START.
LOADING FORMS INTO THE PRINTER

1. Raise the top cover. The release lever is at the bottom front center.

2. Disengage the platen clutch (see illustration for location of referenced carriage controls).

3. Raise the forms check levers. The FORM CHECK light will light.

4. Open the left and right tractor pressure plates.

5. Tip the print position indicator bar back toward you.

6. Set the paper brake tension lever below zero.

7. Turn the platen adjusting knob all the way to the rear (counterclockwise).

8. Feed the paper up and in from the left and lay it back across the carriage.

9. Line up the holes in the paper with the tractor pins and close the tractor pressure plates.

10. Lower the forms check lever. The FORM CHECK light will go out.

11. Set the paper brake tension lever (1 is a good setting for single-part paper).

12. Set the platen adjusting knob (2 is a good setting for single-part paper).

13. Turn the paper advance knob (right side of carriage) until a crease in the paper appears just above the typebar.

14. Close the print position indicator.

LOADING THE CARRIAGE CONTROL TAPE

1. Release the carriage brush holder by lifting up on the unlocking lever.

2. Insert the tape (channel one to the right) and close the brush holder.

3. Press CARRIAGE RESTORE on the operator's panel.

4. Engage the platen clutch.

5. Close the printer top cover.
CHANGING A TYPEBAR

Removing a Typebar

1. Press the 1433 STOP key.
2. Raise the printer top cover.
3. Turn the typebar motor switch (see illustration) to OFF. Allow approximately twenty seconds for the typebar motor to stop.
4. Turn the typebar motor switch to TYPEBAR REMOVAL.
5. Turn the typebar positioning wheel to the right until the end of the typebar shaft appears in the slot in the typebar housing. This slot is directly in front of the typebar motor switch.
6. Holding the shaft lightly in your right hand, exert a gentle pull to the right while continuing to turn the typebar positioning wheel.
7. When the typebar is released draw it straight out to the right.

Inserting a Typebar

1. Holding the typebar so the flag is on the right (typeface toward the machine), guide the typebar straight into its slot in the typebar housing.
2. When the typebar comes to a stop, start turning the typebar positioning wheel to the left until the drive teeth engage the typebar.
3. Position the typebar flag in accordance with the decal on top of the housing.
4. Turn the typebar motor switch ON.
5. Close the printer top cover.
6. Press the 1443 RESET key and then the 1443 START key.
7. A sync check will occur on the first line of print. Press 1443 RESET and START to continue.

2311 DISK STORAGE

IBM 2311 disk storage comprises two major components, the 2311 disk storage drive and the 1316 disk pack.

OPERATOR CONTROLS AND INDICATORS

Start/Stop Switch

Turning the Start/Stop Switch to the Stop position causes the access mechanism to retract from the disk pack and removes power from the disk drive motor. Automatic braking stops disk pack rotation in a few seconds.

Select Lock Indicator

When on, this indicates a machine condition which requires the attention of IBM Customer Engineering. This condition causes the disk storage drive to be disabled and stops the usage meter.
**Enable/Disable Switch**

When the CPU is stopped, this switch enables or disables the communication of the drive with the CPU. It also enables or disables the equipment usage meter.

If the CPU is running when the switch setting is changed, the drive and usage meter operating status are not changed until the CPU is stopped. (See also "Select Lock Indicator".)

**OPERATING PROCEDURES**

The following procedures should be followed for rapid, effective disk pack changing:

**Loading and Readying Procedure.**

- Open the 2311 cover.
- Remove the bottom disk pack cover by turning the bottom locking knob.
- Place the 1316 disk pack (still contained in top cover) on the 2311 spindle.
- Turn the top cover in clockwise direction until firm resistance is met.
- Lift the top cover from the disk pack.
- Close the 2311 cover.
- Set START/STOP switch to START.
- Set ENABLE/DISABLE switch to ENABLE.
- Reassemble the top and bottom covers of the disk pack.
- Store the covers in a clean cabinet or on a clean shelf.
- Turn on the 2841 control unit switch.

**CAUTION:** Do not leave disk pack top cover inside disk drive.

**UNLOADING PROCEDURE**

- Set START/STOP switch to STOP.
- Set ENABLE/DISABLE switch to DISABLE.
- Wait for the disk pack to stop rotating.
- Separate the top and bottom disk pack covers.
- Open the 2311 cover.
- Place the disk pack top cover over disk pack.
- Turn the top cover in direction of OFF arrow at least two full turns.
- Lift the top cover, now containing the disk pack, from the spindle.
- Fasten the bottom cover to disk pack (firmly).
- Close the 2311 cover.
- Store the disk pack in a clean cabinet or on a clean shelf.

**2790 I/O DEVICES**

A comprehensive discussion of the IBM 2791 and 2793 Area Stations, 2795 and 2796 Data Entry Units, and 1035 Badge Reader can be found in IBM 2790 Data Communication System, Component Description, Order Number GA27-3015. The programs that support these I/O devices are discussed in IBM 1800/2790 MPX Data Communication System, Order Number GC26-3732.
Console Operations

Many functions are done at the console by the operator. They are described below.

CLEAR MAIN STORAGE

To clear main storage to zeros:
1. Set the WRITE STOR PROT BITS switch to YES.
2. Set all other console switches OFF.
3. Set the console mode switch to RUN.
4. Press and hold CLEAR STOR.
5. While holding CLEAR STOR, press console START.
6. Release these keys and allow time for main storage to completely cycle.
7. Press console STOP.
8. Press RESET.

To clear storage protect bits only:
1. Set the WRITE STOR PROT BITS switch ON.
2. Set all other console switches OFF.
3. Set the console mode switch to DISPLAY.
4. Press and hold CLEAR STOR.
5. While holding CLEAR STOR, press console START.
6. Release these keys and allow time for main storage to completely cycle.
7. Press console STOP.
8. Return the mode switch to RUN.

Note: If a console check light is on after clearing main storage, reset and repeat the operation.

CONSOLE PROGRAM LOAD (NOT UNDER MPX)

1. Press consoles STOP and RESET.
2. Set the console mode switch to RUN.
3. Place the deck to be loaded into the card read punch hopper and press reader START
4. Press console PROGRAM LOAD.

For a means of loading a program to a location other than zero, see "1442 Relocatable Card Dump Program."
ALTERING OR DISPLAYING THE CONTENTS OF MAIN-STORAGE LOCATIONS

1. With the system stopped, set the console mode switch to LOAD.

2. Set the data switches to the desired four-character hexadecimal main-storage address. Switches 0-3 constitute the first hexadecimal character, 4-7 the second, etc.

3. Press LOAD I (the selected address is displayed in the I-register).

To display the contents of the address:

1. Set the console mode switch to DISPLAY.

2. Press console START.

The contents of the selected location is displayed in the B-register. Successive pressing on the console START key displays consecutive main-storage locations.

To alter the contents of the address:

1. Set the new data word in the console data switches.

2. Set the console mode switch to LOAD.

3. Press console START.

To return to system operation:

1. Set the console mode switch to RUN.

2. Press console START.

OPERATIONS MONITOR

The Operations Monitor comprises an internal program resettable timer and manual controls on the 1800 console. An ALARM indicator on the console will light if the Operations Monitor is in operation and the timer times out. The timer is reset by a CALL OPMON within the prescribed time interval. Once the alarm is activated, it must be manually reset. An audible alarm may also be attached to the Operations Monitor.

PROCEDURE

1. Set the selector switch located on the CE panel below the console to the desired time interval. (Ensure that the call to the Operations Monitor subroutine in the MPX system occurs often enough to prevent the timer from timing out.)

2. With the system to be monitored in operation, turn on the operations Monitor toggle switch on the console. This action initiates the first timer cycle and the system is now being monitored by the Operations Monitor.

If the alarm is activated (program in loop, power failure, etc.), it cannot be reset by the program. You must manually reset it by turning off the Operations Monitor toggle switch on the console.
MPX System Generation

Introduction

System generation is the process of setting up any one of a wide variety of possible MPX systems from materials supplied by IBM Program Information Department (PID).

A system generation must be done whenever a system is to be set up for the first time or whenever the capabilities of a system are to be changed, for example, when a system is to be expanded to support new input/output devices.

Generating Your System

The process of generating your MPX system is accomplished in five major phases:

During phase 1 you will begin the generation by performing a cold start to the disk-resident System Generation Monitor (a small IBM-supplied version of BOM). Cold Start is an MPX disk-resident program whose major functions are to load disk-resident executive programs into main storage and to pass control to them (see "Cold Start"). Once in control, the System-Generation Monitor will cause approximately 2500 blank cards to be punched and passed through the 1442 card read punch. The resultant deck will contain control statements and blank cards necessary to continue the system generation.

During phase 2 you will run the cards punched and passed during phase 1 back through the 1442 card read punch. Sets of cards, called equate cards, will be punched for BOM, the Executive Director, and various Subroutine Library programs. The equate cards are used to define the configuration of your MPX system to be generated. For details, a programmer should refer to the IBM 1800 Multiprogramming Executive Operating System Programmer's Guide, Order Number GC26-3720. Information specifying the various configuration options must be punched into these cards. You should know the values you wish punched into these cards before you begin generating your system. After the equate cards are completed, BOM, the Executive Director, and certain library programs are assembled by the Macro Assembler. Finally, you will define your disk packs and delete certain programs from the PID system generation pack.

During phase 3 you will transfer control from the System Generation Monitor to your BOM which was assembled and punched to cards during phase 2. Under control of your BOM, the System Loader program (punched during phase 2) will be read into main storage. One of the main functions of the System Loader is to process assignment cards, cards used to specify the I/O device and machine function assignments to interrupt levels and bit positions on those levels. For a description of the assignment cards, see Appendix A. During this phase your assignment cards will be processed by the System Loader.

During phase 4 the following batch-processing operations are performed under the control of the Batch-Processing Monitor Supervisor, a disk-resident system program called into execution by your BOM when a // JOB control statement is encountered (see "Background-Processing Operations").

1. Definition of the number of disk drives on your system
2. Labeling of cartridges
3. Definition of main-storage layout
4. Definition of 1810 disk layout

During phase 5, system generation is completed. The previous phase is finished with the Batch-Processing Monitor Supervisor in control. You will now go on to generate a batch-processing system (build BOM) or to generate a real-time system (build the Executive and coreloads).
Note that during all phases of system generation you will be aided by system generation messages and system control messages printed on the list and system printers, respectively.

Materials

In order that you may perform a system generation, the IBM Program Information Department provides the following materials:

CARDS

- An 11-card starter deck. This deck consists of seven cold start loader cards used to load the disk-resident Cold Start program to main storage, a cold start name card used by Cold Start to load and pass control to the System-Generation Monitor (see Cold Start) and three Batch-Processing Monitor control cards used to perform the first phase of system generation.

- A 4-card Basic Operating Monitor (BOM) high core loader. These cards are placed in front of your assembled BOM object deck (phase 3 of system generation). They cause that object deck to be read into main storage and control to be passed to it (see Figure 1).

- An 8-card 1442 relocatable main-storage dump to cards program. It is used as a debugging aid (see "MPX Debugging Aids").

DISKS

- One 1316 disk pack, if your system programs are to reside on a 2311 disk storage drive. The label of your 1316 disk storage pack, sent to you from PID, is PID001.

- Two 2315 disk cartridges, if your system programs are to reside on an 1810 disk storage unit and your system does not include a communications adapter.

- Three 2315 disk cartridges, if your system programs are to reside on an 1810 disk storage unit and your system is to contain one or more communications adapters.

The disk packs and cartridges issued by PID contain all the system programs and data files necessary to generate any MPX system.
Figure 1. BOM High Core Loader Cards
Figure 1. BOM High Core Loader Cards (Cont.)
Materials other than those provided by PID are required to generate a system. You must provide these:

**CARDS**

- 2,600 blank cards (more if you have a single-1810 drive system). These cards will be processed during phase 1 and will become the system-generation input for the other phases. Note that the actual number of cards used will vary between 2,500 and 2,600 depending on which system-generation options are selected.

- Note also that in phases 2 and 3 you will be required to complete equate cards and assignment cards. It is suggested that you know the values you wish punched in these cards prior to beginning the system-generation process.

**SCRATCH DISKS**

One scratch disk is required for each disk received from PID.

**Operating Procedures**

These are two sets of operating procedures for generating your system:

1. Those for generating a system using 2311(s) or two or more 1810 drives, and
2. Those for generating a system using a single 1810 drive.

For generating a system using a single 1810 drive, there is no phase 1 operation, so read the "System Generation Notes" below and skip to the section headed "Phase 2 Operations for Single 1810 Drive System."

For generating a system using 2311(s) or two or more 1810 drives, begin at the section headed "Phase 1 Operations."

**SYSTEM GENERATION NOTES**

1. The successful completion of each DMP operation is indicated by the printing of the message DMP FUNCTION COMPLETED. In the event the operation is unsuccessful, one or more DMP (DXX) error messages may be printed followed by the message DMP FUNCTION ABORTED. See IBM 1800 Multiprogramming Executive Operating System Error Message and Recovery Procedures, Order Number GC26-3727, for the meaning of the message and the procedure to correct the error. When the error has been corrected you may have to revert to a previous step to continue the generation process.

2. Setting program switch 6 ON causes the system to come to a WAIT on any EAC error. See the Error Messages manual for the meaning of the errors and the recovery procedures.

3. Do not stop the system-generating process by pressing the 1442 card punch STOP key. The System Generation Monitor does not contain no-response check subroutines. Pressing the STOP key may cause an interrupt to be lost, thereby placing the system in an interminable loop.

4. System generation may be stopped by pressing console STOP, and later restarted, at any point in the process. After stopping, make certain that all remaining input is saved. Cold start to PID pack 1 or load BOM from cards when restarting the system.
PHASE 1 OPERATIONS

1. Place PID disk 1 (2315 or 1316) on drive 0.

2. Ready drive 0.

3. Set the WRITE STOR PROT BITS switch ON (all other console switches should be OFF).

4. Set the console mode switch to RUN.

5. Clear main storage to zeros (refer to "I/O Device and Console Operations").

6. Place the 11-card starter deck received from PID in the 1442 card read punch.
   
   Note: The cold-start name card must be changed if your 2841 address is not A.

7. Fill the remaining space in the hopper with blank cards (approximately 2,600 cards will be needed during this phase).

8. Ready the card read punch.

9. Set the data switches as follows:
   
   If this system generation uses a 2311 disk storage drive, set:
   
   switch 0 : ON
   switches 1-3 : OFF
   switches 4-7 : OFF
   switches 8-11 : selector channel address (hexadecimal)
   switches 12-15 : OFF
   
   If this system generation does not use a 2311, ensure that all data switches are OFF.

10. Press console PROGRAM LOAD.

11. Ignore any cold start warning messages. Press console START to continue.
   
   Note: Cold Start can be performed only on PID disk 1.

12. The following messages are printed on the system printer:

   IBM 1800 DACS MPX/SYS GEN MON 00,000
   SEN SW 0 ON ABSOLUTE LOADER
   SEN SW 1 ON LOAD BP MONITOR

13. Set sense switch 1 ON.

14. Press console START.

15. The following messages are printed on the system printer and the blank cards are processed through the 1442 card read punch.

   // JOB 00001 00 JAN 00 00,000 HRS
   // DMP 00 JAN 00 00,000 HRS
   *SRFLE P GENCD PUNCHING SYS GEN CONTROL CARDS

   The control cards required to generate MPX are punched. These cards will be the input for phase two of the system-generation process. Note that some cards are processed without being punched. Keep removing cards from the stacker, placing them into a container in the order in which they are punched. When the process is complete, the number of cards processed is indicated by a message.

   002520 CARDS PROCESSED
   DMP FUNCTION COMPLETED

   Blank cards remaining in the hopper are passed to the second stacker without processing. Ignore any card reader not ready message.

16. Remove the 11-card starter deck from the front of the processed output.
This completes phase 1. If you are generating a system using 2311 drive(s) or two or more 1810 drives, skip to the section headed "Phase 2 Operations for 2311 or Two or More 1810 Drive Systems."

PHASE 2 OPERATIONS FOR SINGLE 1810 DRIVE SYSTEM

The following is the operating procedure for generating a system using a single 1810 drive:

1. Prepare the following deck (comments cards are optional):

// JOB 00001
// *
// * PRESS START IF SYSTEM HAS A 1443 PRINTER
// *
// * PRESS CONSOLE INTERRUPT WITH SENSE SWITCH 7 UP IF SYSTEM
// * DOES NOT HAVE A 1443 PRINTER
// *
// PAUSE
// *
// * ALL PRINTING WILL BE DONE ON 1443 FROM NOW ON
// *
// SET A 0 0
// JOB 00001
// *
// THIS PASS GENERATES THE MPX SYSTEM. FOLLOW ALL PRINTED
// INSTRUCTIONS.
// *
// * A DUMP OF BOM, EXDIR, AND LIBRARY EQUATE CARDS FOLLOWS.
// * THE USER MUST PUNCH
// * AN ENTRY IN THE OPERAND FIELD BEFORE RELOADING THESE CARDS.
// * A SRFLE END CARD MUST BE INCLUDED WITH EQU’S.
// ********************************************************************
// DMP
*SRFLE P EQUEX PUNCHING EXDIR EQU CARDS******

| (insert 130 blank cards) |
// PAUSE REMOVE AND DEFINE 'EXDIR' EQU CARDS
// DMP *SRFLE P EQUBM PUNCHING BOM EQU CARDS*********

| (insert 220 blank cards) |
// PAUSE REMOVE AND DEFINE 'BOM' EQU CARDS
// DMP *SRFLE P EQULB PUNCHING LIBRARY EQU CARDS****

| (insert 75 blank cards) |
// * THE EQUATE CARDS FOR VARIOUS SUBROUTINES ARE SEPARATED FROM EACH OTHER
// * BY CARDS CONTAINING ONLY ASTERISKS, SEE OPERATING PROCEDURES, SYSTEM
// * GENERATION, EQUATE CARDS.
// *
// PAUSE REMOVE AND DEFINE 'LIBR'EQU CARDS
// JOB 00001
// *
// * A DUMP OF SYSTEM LOADER AND BOM CARD UTILITIES FOLLOWS. AT EACH
// * PAUSE, REMOVE CONTROL CARDS FROM PUNCHED OUTPUT AND LABEL DECK.
// *
// PRESS CONSOLE START TO CONTINUE.
// *
// DMP *DUMP UAO PN SYSLD IS BEING PUNCHED*********

| (insert 120 blank cards) |
// PAUSE REMOVE AND LABEL 'SYSTEM LOADER'
// DMP *DUMP UAO PN BDMAP IS BEING PUNCHED**********
I (insert 45 blank cards)

// PAUSE REMOVE AND LABEL 'BOM DISK WRITE ADDRESS PROGRAM'
// DMP *DUMP UAO PN BDUPL IS BEING PUNCHED

I (insert 50 blank cards)

// PAUSE REMOVE AND LABEL 'BOM DISK DUPLICATION PROGRAM'
// DMP *DUMP UAO PN BDPAT IS BEING PUNCHED

I (insert 35 blank cards)

// PAUSE REMOVE AND LABEL 'BOM DISK PATCH PROGRAM'
// DMP *DUMP UAO PN BLIST IS BEING PUNCHED

I (insert 30 blank cards)

// PAUSE REMOVE AND LABEL 'BOM 80-80 LIST PROGRAM'
// DMP *DUMP UAO PN BDCRL IS BEING PUNCHED

I (insert 40 blank cards)

// PAUSE REMOVE AND LABEL 'BOM DISK DUMP AND RELOAD PROGRAM'
// DMP *DUMP UAO PN BDPIP IS BEING PUNCHED

I (insert 90 blank cards)

// PAUSE REMOVE AND LABEL 'BOM 2311 DISK PACK INITIALIZATION PROGRAM'
// DMP *DUMP UAO PN BDUMP IS BEING PUNCHED

I (insert 25 blank cards)

// PAUSE REMOVE AND LABEL 'BOM 1810 DISK DUMP TO CARDS'
// JOB 00001
// DMP *SRFLE P SAMPL IS BEING PUNCHED

I (insert 35 blank cards)

// PAUSE REMOVE AND LABEL 'BOM 1810 DISK RELOAD FROM CARDS'
// DMP *DUMP DKO PN

// PAUSE REMOVE AND LABEL 'THE MPX SAMPLE PROGRAM'
// * NOW DUMP PID PACK 1 TO CARDS FOR BACKUP
// * DMP *DUMP DKO PN

24 1800 MPX Operating Procedures
// PAUSE  SAVE FOR BACKUP
// **
// * AT PAUSE PLACE USER COMPLETED LIBRARY EQUATE CARDS BEHIND RESPECTIVE
// * SRFLE M CONTROL CARDS FOR EXDCD,CSPCD,LSPCD,MDFCD,MFICD,ADRCD,DMPSC.
// * PRESS READER AND CONSOLE START TO CONTINUE.
// *
// * BE SURE TO INCLUDE A SRFLE END CARD AT THE END OF EACH
// * SET OF EQUATES.
// *
// PAUSE
// JOB  00001

// DMP
*SRFLE M  EXDCD  EXECUTIVE EQUATE CARDS MUST FOLLOW
*SRFLE M  CSPCD  CSPAR EQUATE CARDS MUST FOLLOW
*SRFLE M  LSPCD  LSPCL EQUATE CARDS MUST FOLLOW
*SRFLE M  MDFCD  MDFIO EQUATE CARDS MUST FOLLOW
*SRFLE M  MFICD  MFIO  EQUATE CARDS MUST FOLLOW
*SRFLE M  ADRCD  ADRCK EQUATE CARDS MUST FOLLOW
*SRFLE M  DMPSC  DMPS EQUATE CARDS MUST FOLLOW
// JOB  00001

// ** ASSEMBLE EXECUTIVE DIRECTOR, PUNCH THEN STORE.
// ASM EXDIR EXDCD

*STORE  EXDIR
// *
// ** THE FOLLOWING CONTROL CARDS ASSEMBLE AND STORE THE SUBROUTINES
// ** AFFECTED BY THE JUST-INSERTED LIBRARY EQUATE CARDS.
// *
// * ASM ADRCK ADRCD
*STORE  0  ADRCK
// ASM DMPS  DMPSC
*STORE  0  DMPS
// ASM CSPAR  CSPCD
*STORE  0  CSPAR
// ASM LSPCL  LSPCD
*STORE  0  LSPCL
// ASM MDFIO  MDFCD
*STORE  0  MDFIO
// ASM MFIO  MFICD
*STORE  0  MFIO
// JOB  00001

// ** THE FOLLOWING CONTROL CARDS WILL DELETE THE ENTIRE COMMUNICATION
// ** OBJECT LIBRARY. IF THIS IS DESIRED, PRESS START.
// ** IF IT IS NOT DESIRED TO DELETE THE ENTIRE COMMUNICATION
// ** LIBRARY, THEN SET PROGRAM SWITCH 7, PRESS CONSOLE INTERRUPT
// ** TO CONTINUE.
// *
// PAUSE

// DMP
*DELETE R  BSCTR  CA TRACE ROUTINE
*DELETE  BSCTR  CA TRACE ROUTINE
*DELETE R  BSCCK  I/O AREA CHECK ROUTINE
*DELETE R  ZIPPCD  CONVERSION ROUTINE
*DELETE  ZIPPCD  CONVERSION ROUTINE
*DELETE  USHOL  USASCII TO HOLLERITH TBL
*DELETE  HOLUS  HOLLERITH TO USASCII CODE TBL
*DELETE  USEBC  USASCII TO EBCDIC TABLE
*DELETE  EBCUS  EBCDIC TO USASCII TABLE
*DELETE  USPRT  USASCII TO 1443 PRINTER TABLE
*DELETE  USTYP  USASCII TO TYPEWRITER TABLE
*DELETE R  PEFIB  PACKED EBCDIC TO FIXED BINARY
*DELETE R  PEFIB  PACKED EBCDIC TO FIXED BINARY
*DELETE R  FLBPE  FLOATING BINARY TO PACKED EBCDIC
*DELETE R  FLBPE  FLOATING BINARY TO PACKED EBCDIC
// JOB  00001

// ** IF THE LIBRARY SUBROUTINES TO SUPPORT THE USE OF 2311'S ARE NOT
// ** NEEDED PRESS START. OTHERWISE, PRESS CONSOLE INTERRUPT WITH
// ** SENSE SWITCH 7 ON.
// PAUSE
// DMP
*DELET  R  DSOR
*DELET  R  DSCR
*DELET  R  MDAT
*DELET  R  DPPI
*DELET  R  RESFR
*DELET  R  USPRO
// JOB  00001
// * THE FOLLOWING SOURCE FILES AND UTILITIES ARE DELETED
// * DELETE-- EQUBM, EQUEX, EQULB, GENCD, CSPCD, LSPCD, MDFCD, MFICD, EXDCD
// * BLIST, BDWAP, SYSLD, BDWAP, BDAT, BDPIP, BDCRL, ADRCD, DMPSC, SAMP1
// * IF ALL PRECEDING OPERATIONS COMPLETED SUCCESSFULLY,
// * PRESS CONSOLE START TO CONTINUE.
// *
// PAUSE  PRESS START TO CONTINUE
// DMP
*DELET  D  EQUEX
*DELET  D  CSPAR
*DELET  D  LIBRARY EQUATES
*DELET  D  CSPCD
*DELET  D  LSPCD
*DELET  D  MDFCD
*DELET  D  MFICD
*DELET  D  EXDCD
*DELET  R  BLIST
*DELET  R  BDWAP
*DELET  R  SYSLD
*DELET  R  BDUPL
*DELET  R  BDAT
*DELET  R  BDPIP
*DELET  R  BDCRL
*DELET  R  BDUMP
*DELET  R  BRELD
*DELET  R  ADRCD
*DELET  D  DMPSC
*DELET  R  SAMP1
// JOB
// *
// * THE FOLLOWING CONTROL CARDS DELETE THE 2790 OBJECT LIBRARY
// *
// DMP
*DELET  R  CLIP
*DELET  R  CLTP
*DELET  R  FILFS
*DELET  R  LACCN
*DELET  R  LINKF
*DELET  R  LOOP1
*DELET  R  LOOP2
*DELET  R  PULSE
*DELET  R  WRTLN
// JOB
// *
// DEFINE PAKDK 0
// JOB  00001
// *
// * MOUNT PID PACK 2 ON DRIVE 0
// * DO NOT COLD START.
// *
// PAUSE
// JOB  00002
// *
// * NOW DUMP PID PACK 2 TO CARDS FOR BACKUP
// *
// DMP
*DUMP  DKO PN
// PAUSE SAVE FOR BACKUP

// *

// * AT PAUSE PLACE USER-COMPLETED EQUATE CARDS BEHIND

// * SRFLE M CONTROL CARD FOR BOM

// * PRESS READER AND CONSOLE START TO CONTINUE.

// *

// PAUSE

// *

// DMP

*SRFLE M	 BOMCD

// *

// NOW ASSEMBLE AND PUNCH BOM

// *

// ASM BOM	 BOMCD

// *

// AT PAUSE MOUNT PID PACK 1 IF SYSTEM DOES NOT REQUIRE CA.

// * MOUNT PID PACK 3 IF SYSTEM REQUIRES CA.

// *

// PAUSE

// *

// JOB	 00003

// *

// SINCE THE ONE DRIVE SYSTEM DOES NOT SUPPORT THE 2790 ADAPTER,

// DELETE THE FOLLOWING FILE.

// *

// DMP

*DELET	 D	 X2790

// *

// NOW DUMP PID PACK 3 TO CARDS FOR BACKUP

// *

// DMP

*DUMP	 DK0 PN
// PAUSE  SAVE FOR BACKUP
// *
// *  AT PAUSE PLACE BSCIO SYSGEN EQUATES AFTER *SRFLE CONTROL CARD
// *  PRESS READER START AND CONSOLE START TO CONTINUE
// *
// PAUSE
// *
// DMP
*SRFLE M  BSCCD
// *
// *
// ASSEMBLE AND PUNCH BSCIO
// *
// ASM BSCIO BSCCD

* THE EQUATE CARDS FOR VARIOUS SUBROUTINES ARE SEPARATED FROM EACH OTHER
// *  BY CARDS CONTAINING ONLY ASTERISKS, SEE OPERATING PROCEDURES, SYSTEM
// *  GENERATION, EQUATE CARDS.
// *
// PAUSE  REMOVE AND DEFINE 'LIBR'EQU CARDS
// JOB  00001
// *
// *  A DUMP OF SYSTEM LOADER AND BOM CARD UTILITIES FOLLOWS. AT EACH
// *  PAUSE, REMOVE CONTROL CARDS FROM PUNCHED OUTPUT AND LABEL DECK.
// *  PRESS CONSOLE START TO CONTINUE.
// *
// DMP
*DUMP  UAO PN SYSLD IS BEING PUNCHED******************************

* REMOVE AND DEFINE 'LIBR'EQU CARDS
// JOB  00001
// *
// *  A DUMP OF SYSTEM LOADER AND BOM CARD UTILITIES FOLLOWS. AT EACH
// *  PAUSE, REMOVE CONTROL CARDS FROM PUNCHED OUTPUT AND LABEL DECK.
// *  PRESS CONSOLE START TO CONTINUE.
// *
// DMP
*DUMP  UAO PN BDWAP IS BEING PUNCHED******************************

* REMOVE AND DEFINE 'LIBR'EQU CARDS
// JOB  00001
// *
// *  A DUMP OF SYSTEM LOADER AND BOM CARD UTILITIES FOLLOWS. AT EACH
// *  PAUSE, REMOVE CONTROL CARDS FROM PUNCHED OUTPUT AND LABEL DECK.
// *  PRESS CONSOLE START TO CONTINUE.
// *
// DMP
*DUMP  UAO PN BDUPL IS BEING PUNCHED******************************

* REMOVE AND DEFINE 'LIBR'EQU CARDS
// JOB  00001
// *
// *  A DUMP OF SYSTEM LOADER AND BOM CARD UTILITIES FOLLOWS. AT EACH
// *  PAUSE, REMOVE CONTROL CARDS FROM PUNCHED OUTPUT AND LABEL DECK.
// *  PRESS CONSOLE START TO CONTINUE.
// *
// DMP
*DUMP  UAO PN BDPAT IS BEING PUNCHED******************************

* REMOVE AND DEFINE 'LIBR'EQU CARDS
// JOB  00001
// *
// *  A DUMP OF SYSTEM LOADER AND BOM CARD UTILITIES FOLLOWS. AT EACH
// *  PAUSE, REMOVE CONTROL CARDS FROM PUNCHED OUTPUT AND LABEL DECK.
// *  PRESS CONSOLE START TO CONTINUE.
// *
// DMP
*DUMP  UAO PN BLIST IS BEING PUNCHED******************************
This entire file will be input to phase 2 of the system-generation process for a single-1810 drive system.

2. Place PID disk 1 on drive 0.

3. Ready drive 0.

4. Set the WRITE STOR PROT BITS switch ON (all other console switches should be OFF).

5. Set the console mode switch to RUN.

6. Clear main storage to zeros (refer to "I/O Device and Console Operations").

7. Place the 7-card cold start loader and the cold start name card received from PID in the 1442 card read punch.

8. Fill the remaining space in the hopper with the system-generation input file that you punched in step 1.

9. Ready the card read punch.
10. Press console PROGRAM LOAD.

11. Ignore any cold start warning messages. Press console START to continue.

   Note: Cold Start can be performed only on PID disk 1.

12. The following messages are printed:

   IBM 1800 DACS MPX/SYS GEN MON 00.000
   SEN SW 0 ON ABSOLUTE LOADER
   SEN SW 1 ON LOAD BP MONITOR

13. Set sense switch 1 ON.

14. Press console START.

15. Execute steps 3 through 13 under the section headed "Phase 2 Operations for 2311 or Two or More 1810 Drive Systems" and return.

16. You will now dump PID disk 1 to cards to save it for a backup. You can reload it by using BRELDD. Press console START to continue.

17. Execute steps 75 through 81 under the section headed "Phase 2 Operations for 2311 or Two or More 1810 Drive Systems" and return. Note that this procedure will differ slightly in that BOM equates are not placed on disk by SRFLE nor is BOM assembled.

18. Execute steps 85 through 87 under the section headed "Phase 2 Operations for 2311 or Two or More 1810 Drive Systems" and return.

19. You will now dump PID disk 2 to cards to save it for backup. Place PID disk 2 on drive 0.

20. Ready drive 0.

21. Press console START to continue (to dump PID disk to cards).

22. Place your completed BOM equate cards behind the *SRFLE card in the 1442.

23. Ready the 1442.

24. Press console START to continue. The BOM equate cards are now read to disk and BOM is assembled.

25. Remove and label the BOM object deck (just punched).

26. If your system does not require communications adapters, skip to step 33. If your system does require one or more communications adapters, mount PID disk 3 on drive 0.

27. Ready drive 0.

28. Press console START to continue. You will not dump PID disk 3 to cards to save for backup.

29. Place your completed BSCIO equate cards behind the *SRFLE card in the 1442.

30. Ready the 1442.

31. Press console START to continue. The BSCIO equate cards are now read to disk and BSCIO is assembled.

32. Remove and label the BSCIO object deck (just punched).

33. Place PID disk 1 on drive 0.

34. Ready drive 0.

35. Press console START.
36. If your system does not contain any communications adapters, set program switch 7 ON and press CONSOLE INTERRUPT. This completes phase 2 operations for a single-1810 drive system. To continue, go to the section headed "Phase 3 Operations." If your system does contain one or more communications adapters, press console START.

37. Place the BSCIO object deck punched in step 31 behind the *STORE card in the 1442.

38. Ready the 1442.

39. Press console START to continue. The BSCIO program is now stored to disk.

This completes phase 2 operations for a single-1810 drive system. To continue, go to the section headed "Phase 3 Operations."

PHASE 2 OPERATIONS FOR 2311 OR TWO OR MORE 1810 DRIVE SYSTEMS

1. Fill the 1442 card read punch with the cards processed during phase 1 operations.

2. Ready the 1442 card read punch.

3. The following messages are printed on the system printer:

   // JOB 00001 00 JAN 00 00,000 HRS
   // *
   // * PRESS START IF SYSTEM HAS A 1443 PRINTER
   // *
   // * PRESS CONSOLE INTERRUPT WITH SENSE SWITCH 7 UP IF SYSTEM
   // * DOES NOT HAVE A 1443 PRINTER
   // *
   // PAUSE 00 JAN 00 00,000 HRS

4a. If your system has a 1443 printer, ready it and press console START. The following messages are printed on the system printer:

   CONTINUE
   // *
   // * ALL PRINTING WILL BE DONE ON 1443 FROM NOW ON
   // *
   // SET A 0 0 00 JAN 00 00,000 HRS
   IN ST LIST SYS
   OLD A 1 1
   NEW A 0 0

   and on the 1443 printer:

   // SET A 0 0 00 JAN 00 00,000 HRS
   IN ST LIST SYS
   OLD A 1 1
   NEW A 0 0

   Go to step 5.

4b. If your system does not have a 1443 printer, set sense switch 7 ON and press CONSOLE INTERRUPT.

5. The following messages are printed and the equate cards for the Executive Director are punched. Some of the following //JOB cards may be different if your system uses 1810 drives.
6. When the program pauses, remove and label the Executive Director equate cards. Note that control cards precede the equate cards (cards with no sequence numbers in columns 73-80) in the stacker. Disregard these cards as they are no longer needed.

7. Press console START to continue. The following messages are printed and the equate cards for BOM are punched.

CONTINUE
// DMP 00 JAN 00 00.000 HRS
*SRFLE P EQUBM
000217 CARDS PROCESSED
DMP FUNCTION COMPLETED
// PAUSE REMOVE AND LABEL 'BOM' EQU CARDS 00 JAN 00 00.000 HRS

8. When the program pauses, remove and label the BOM equate cards. Omit control cards.

9. Press console START to continue. The following messages are printed and equate cards for various Subroutine Library programs are punched.

CONTINUE
// DMP 00 JAN 00 00.000 HRS
*SRFLE P EQULB
000031 CARDS PROCESSED
DMP FUNCTION COMPLETED
// PAUSE REMOVE AND LABEL 'LIBR' EQU CARDS 00 JAN 00 00.000 HRS

10. When the program pauses, remove and label the deck of LIBR equate cards. Omit the control cards which precede and follow the equate cards in the stacker. Cards containing only asterisks separate the equate cards for the various subroutines. Replace these cards with cards containing 9s in the last eight columns.

11. Press console START to continue. The following messages are printed and the System Loader is punched to cards.

CONTINUE
// JOB 00001 00 JAN 00 00.000 HRS
// *
// * A DUMP OF SYSTEM LOADER AND BOM CARD UTILITIES FOLLOWS. AT EACH
// * PAUSE, REMOVE CONTROL CARDS FROM PUNCHED OUTPUT AND LABEL DECK.
// * PRESS CONSOLE START TO CONTINUE.
// *
// DMP 00 JAN 00 00.000 HRS
*DUMP UAO PN SYSLD IS BEING PUNCHED***************
DMP FUNCTION COMPLETED
// PAUSE REMOVE AND LABEL 'SYSTEM LOADER' 00 JAN 00 00.000 HRS

12. When the program pauses, remove and label the System Loader deck. Omit all control cards.

13. Press console START to continue. The following sets of messages will be printed and card decks punched. At each pause between the decks, remove control cards from the punched output and label the decks. Press console START after each to continue.
At this point the system-generation procedure differs depending on which types of disk units you have in your system.

If your system uses physical 1810 drives for this system generation, the following messages are printed:

```
PLACE SCRATCH CARTRIDGE ON DRIVE 1. ADDRESS SCRATCH CARTRIDGE
USING JUST-PUNCHED BOM DISK WRITE ADDRESS PROGRAM BDWAP.
WHEN MPX/BOM MESSAGES ARE PRINTED-- NPRO READER, PLACE BDWAP DECK IN READER, FOLLOW WITH NPRO JOB CARD AND REMAINDER OF STACKED INPUT, READY READER, TURN ON RS 0, DATA SWITCH 15, AND PRESS CONSOLE START.
LEAVE DATA SWITCH 15 ON FOR ONE TRY. PRESS CONSOLE START.
LEAVE DATA SWITCH 15 ON FOR DRIVE 1. PRESS CONSOLE START.
AT COMPLETION OF BDWAP, TURN OFF CONSOLE SWITCHES, PRESS STOP, RESET AND START, THEN SET PROGRAM SWITCH 7 ON, AND PRESS CONSOLE INTERRUPT TO CONTINUE.
END 00 JAN 00 00.000 HRS
```
If your system uses 2311 drives for this system generation, the following messages are printed:

CONTINUE
/// *
/// * IF THIS SYSTEM GENERATION USES A SINGLE 2311 DRIVE, IGNORE THE INSTRUCTIONS TO THE NEXT END CARD.
/// *
/// * PLACE A SCRATCH PACK ON DRIVE 1. INITIALIZE IT USING THE JUST-PUNCHED BOM 2311 DISK PACK INITIALIZATION PROGRAM (BDPIP) AND COPY ONTO IT THE CONTENTS OF DRIVE 0 USING THE BOM DISK DUPLICATION PROGRAM (BDUPL), AS FOLLOWS
/// *
/// * WHEN MPX/BOM MESSAGES ARE PRINTED— NPRO READER, PLACE BDPIP DECK IN READER. PLACE BDPIP CONTROL CARDS AFTER OBJECT DECK. NEXT PLACE BDUPL, THE NPRO CARDS AND THE REMAINDER OF THE STACKED INPUT IN THE READER. READY THE READER, TURN ON SS 0, AND DATA SWITCH 15. PRESS CONSOLE START. AS MESSAGES ARE PRINTED, YOU MAY PRESS START TO CONTINUE.
/// *
/// * AT COMPLETION OF BDPIP, PRESS CONSOLE START WITH SSO AND DATA SWITCH 15 ON.
/// *
/// * TO COPY DRIVE 0 TO DRIVE 1, REFER TO OPERATING PROCEDURES MANUAL FORM C26-3725, BOM CARD UTILITIES, FOR DETAILS.
/// *
/// * SAVE COPY OF PID PACK
/// *
/// * AT COMPLETION OF BDUPL, TURN OFF CONSOLE SWITCHES, PRESS STOP, RESET AND START, THEN SET PROGRAM SWITCH 7 UP, AND PRESS CONSOLE INTERRUPT TO CONTINUE.
/// *
/// END 00 JAN 00 00.000 HRS

The following messages are then printed (in both cases) on the system printer:

IBM 1800 DACS MPX/SYS GEN MON 00.000
SEN SW 0 ON ABSOLUTE LOADER
SEN SW 1 ON LOAD BP MONITOR

If your system generation uses 2315 disk cartridges, go to step 15 to continue.

If your system generation uses a 1316 disk pack and your system has more than one 2311 disk drive, skip to step 58 to continue.

If your system generation uses a 1316 disk pack and your system has only one 2311 disk drive, skip to step 46 to continue.

SYSTEM GENERATION USING 2315 DISK CARTRIDGES

15. Load your scratch cartridge to drive 1.


17. Remove all cards from the 1442 hopper.

18. Nonprocess out all the cards inside the 1442. Omit all these cards except the last which is a // JOB card.

19. Place the BOM Disk Write Addresses Program, BDWAP (one of the BOM utilities punched in step 13), into the 1442 hopper.

20. Follow it with the // JOB card and the remainder of the system-generation input deck from steps 17 and 18.
21. Ready the 1442 card read punch.

22. Set sense switch 0 ON and data switch 15 ON. Press console START.

23. The following messages are printed:

**BOM DISK WRITE ADDRESSES PROGRAM**
Enter NO TRIES ON DATA SW MAX 001F

Enter the number of tries in the data switches (/0 xx) (described under "BDWAP - BOM Disk Write Addresses Program") and press console START.

24. The following messages are printed:

**DATA SWITCHES EQUAL LOGICAL DRIVE**

```
DRIVE CODES - HEX 0000 0001 0002
```

Press console START.

25. The following message is printed:

**DRIVE SELECTED IS 1. IF CORRECT, PRESS START WITH SENSE SWITCH 0 ON**

Press console START.

26. The following messages should be printed:

**THERE ARE NO DEFECTIVE CYLINDERS**

**BOM DISK WRITE ADDRESSES PROGRAM**
Enter NO TRIES ON DATA SW MAX 001F

If a message indicating that there are defective cylinders is printed, refer to "BOM Card Utilities," BOM Disk Write Address Program (BDWAP). If the cartridge is defective, replace the cartridge and return to step 15.

27. Set all data switches OFF.

28. Press console STOP, RESET and START.

29. Set program switch 7 ON and press CONSOLE INTERRUPT. The following messages will be printed:

```
// JOB  00001  00 JAN 00 00,000 HRS
// DMP  00 JAN 00 00,000 HRS
*DCOPY 0 1
DMP FUNCTION COMPLETED
// *
// * SAVE PID PACK 1, REMOVE COPY OF PID PACK1 FROM DRIVE 1 AND SAVE.
// * MOUNT PID PACK2 ON DRIVE 0 AND SCRATCH PACK ON DRIVE 1.
// * DO NOT COLD START.
// *
// *
// *
// *
// *
// *
// *
// *
```

WHEN MPX/BOM MESSAGES ARE PRINTED-- NPRO READER, PLACE BDWAP DECK IN READER, FOLLOW WITH NPRO JOB CARD AND REMAINDER OF STACKED INPUT, READY READER, TURN ON SS 0, DATA SWITCH 15, AND PRESS CONSOLE START.

```
// *
// *
// *
// *
// *
// *
// *
// *
// *
// *
// *
// *
// *
// *
// *
// *
```

LEAVE DATA SWITCH 15 ON FOR ONE TRY. PRESS CONSOLE START.

```
// *
// *
// *
// *
// *
// *
// *
// *
// *
```

AT COMPLETION OF BDWAP, TURN OFF CONSOLE SWITCHES, PRESS STOP, RESET AND START. THEN SET PROGRAM SWITCH 7, AND PRESS CONSOLE INTERRUPT TO CONTINUE.

```
// *
// *
```

END  00 JAN 00 00,000 HRS
The following is then printed on the system printer:

IBM 1800 DACS  MPX/SYS GEN MON  00.000
SEN SW 0 ON ABSOLUTE LOADER
SEN SW 1 ON LOAD BP MONITOR

30. Unload and save cartridges from drives 0 and 1. (PID pack 1 master and its copy.)

31. Mount PID pack 2 on drive 0 and a scratch pack on drive 1.

32. Write disk addresses on scratch pack (drive 1) by repeating steps 15 through 29. The BDWAP card deck should be removed from the 1442 and saved for future use.

33. When you have redone those steps, the following messages will be printed:

// JOB 00002 00 JAN 00 00.000 HRS
// DMP 00 JAN 00 00.000 HRS
*DLABL 1 11112
DMP FUNCTION COMPLETED

This labels drive 1 with the label 11112.

// JOB 0000211112 00 JAN 00 00.000 HRS
// DMP 00 JAN 00 00.000 HRS
*DEFINE CONFIG C1 163
DMP FUNCTION COMPLETED

This defines a Core Image Area on drive 1.

// JOB 0000211112 00 JAN 00 00.000 HRS
// *
// * AT PAUSE PLACE USER-COMPLETED EQUATE CARDS BEHIND
// * SRFLE M CONTROL CARD FOR BOM
// * PRESS READER AND CONSOLE START TO CONTINUE.
// *
// DMP 00 JAN 00 00.000 HRS
*DFILE 1 BOM 1304
WILL RESERVE AT SCTR ADDR 1128
DMP FUNCTION COMPLETED

// PAUSE 00 JAN 00 00.000 HRS

34. Remove all cards from the 1442 hopper.

35. Nonprocess out all the cards inside the 1442. Omit all these cards except the // DMP and *SRFLE control cards.

36. At this point you should complete your BOM, Executive Director and various Subroutine Library program equate cards (if this has not already been done).

INSERTING EQUATE CARDS

37. Place your completed BOM equate cards behind the *SRFLE card from step 35. Place these cards in the 1442 hopper. Fill the hopper with the remaining system-generation input deck from step 34.
38. Press console START. The following messages will be printed:

CONTINUE
// DMP 00 JAN 00 00,000 HRS
*SRFLE M BOMCD BOM
016548 CARDS ON FILE
001199 SECTORS USED
000105 SECTORS AVAIL
000217 CARDS READ
DMP FUNCTION COMPLETED
// *
// * REMOVE PID CARTRIDGE 2 FROM DRIVE 0 AND SAVE.
// * MOUNT COPY OF PID CARTRIDGE 1 ON DRIVE 0.
// *
// * NEXT PUSH IMMEDIATE STOP AND START TO RELOAD SGMON.
// *
// * THEN WITH SENSE SWITCH 1 ON PUSH START TO LOAD BPMON.
// PAUSE 00 JAN 00 00,000 HRS

39. Unload the PID cartridge 2 from drive 0.

40. Load drive 0 with a copy of PID cartridge 1.

41. Press console IMMEDIATE STOP and START. The following messages are printed on the system printer:

IBM 1800 DACS MPX/SYS GEN MON 00,000
SEN SW 0 ON ABSOLUTE LOADER
SEN SW 1 ON LOAD BP MONITOR

42. Set sense switch 1 ON. Press console START. The following messages are printed:

// JOB 00001 00 JAN 00 00,000 HRS
// *
// * AT PAUSE PLACE USER COMPLETED LIBRARY EQUATE CARDS BEHIND RESPECTIVE
// * *SRFLE M CONTROL CARDS FOR CSPCD,LSPCD,MDFCD,MFICD,ADRCD,DMSC,EXDCD.
// * PRESS READER AND CONSOLE START TO CONTINUE.
// *
// * BE SURE TO INCLUDE A SRFLE END CARD AT THE END OF EACH
// * SET OF EQUATES.
// *
// PAUSE 00 JAN 00 00,000 HRS

43. Place your completed Executive Director and Subroutine Library equate cards behind their respective *SRFLE control cards. The *SRFLE control cards are in the system-generation input file in the 1442 card read punch. Be sure that you have included cards containing 9s in the last eight columns behind each set of equates (refer to step 10).

The last set of equates in the library equate deck are for BSCIO and will be placed behind the proper *SRFLE control card in a later step. Mark these equates as BSCIO equates and set them aside.

44. Ready the 1442.
45. Press console START. The following messages are printed indicating the processing of the equate cards by the SRFLE function:

```
CONTINUE
// JOB 000001 00 JAN 00 00.000 HRS
// DMP 00 JAN 00 00.000 HRS
*SRFLE EXECUTIVE EQUATE CARDS MUST FOLLOW
m
EXOCD EXECUTIVE EQUATE CARDS MUST FOLLOW
CARDS READ 000005
SECTORS AVAIL 000008
DMP FUNCTION COMPLETED
000003
000003
000003
000003
000003
000003
*SRFLE M CSPCD CSPAR EQUATE CARDS MUST FOLLOW
CARDS READ 000005
SECTORS AVAIL 000003
DMP FUNCTION COMPLETED
000003
000003
000003
000003
000003
000003
*SRFLE M LSPCD LSPCL EQUATE CARDS MUST FOLLOW
CARDS READ 000005
SECTORS AVAIL 000003
DMP FUNCTION COMPLETED
000003
000003
000003
000003
000003
000003
*SRFLE M MFICD MFIO EQUATE CARDS MUST FOLLOW
CARDS READ 000005
SECTORS AVAIL 000002
DMP FUNCTION COMPLETED
000003
000003
000003
000003
000003
000003
*SRFLE M ADRCI ADRCK EQUATE CARDS MUST FOLLOW
CARDS READ 000002
SECTORS AVAIL 000002
DMP FUNCTION COMPLETED
000003
000003
000003
000003
000003
000003
*SRFLE M DMPSC DMP EQUATE CARDS MUST FOLLOW
CARDS READ 000002
SECTORS AVAIL 000002
DMP FUNCTION COMPLETED
000003
000003
000003
000003
000003
000003
```

Go to "Assembly of System Programs for a System with 2315 Disk Cartridges."

**SYSTEM GENERATION USING A 1316 DISK PACK**

Because your system generation uses a 1316 disk pack and your system has only one 2311 disk drive, you will now produce a copy of your PID pack on cards.

**DUMPING THE PID PACK TO CARDS**

46. Remove all cards from the 1442 hopper.

47. Nonprocess out all the cards inside the 1442. Omit all those cards except the last, which is a // JOB card.

48. Place the BOM Disk Dump and Reload program, BDCRL (one of the BOM utilities punched in step 13), into the 1442 hopper.

49. Dump cylinder 0 to cards (refer to "BOM Card Utilities").

50. Remove the BDCRL card deck from the 1442 stacker and save it.

51. Remove the cards containing the dump of the PID pack and save it.
52. Punch the following statements into cards and place those cards in the 1442 hopper:

// JOB 0000100002
// DMP
*DUMP DK0 PN
*DUMP DK1 PN

53. Follow each of the *DUMP statements by approximately 12,000 cards. You will now dump the mapped 1810 drives to cards.

54. Set sense switch 7 ON.

55. Ready the 1442 card read punch.

56. Press CONSOLE INTERRUPT.

57. When the dump operations have been completed, save the cards removed from the 1442. Skip to step 71.

DUPLICATING THE PID DISK PACK

Because your system has two or more 2311 drives, you will now copy the PID pack to a scratch pack.

58. Load your scratch pack on drive 1.

59. Remove all cards from the 1442 hopper.

60. Nonprocess out all cards inside the 1442. Omit all those cards except the last which is a // JOB card.

61. Place the BOM 2311 Disk Pack Initialization Program, BDPIP (one of the BOM utilities punched in step 13), into the 1442 hopper.

62. Follow BDPIP in the 1442 hopper by control cards necessary to initialize the disk pack on drive 1 (refer to "BOM Card Utilities").

63. Follow the BDPIP control cards by the BOM Disk Duplication Program, BDUPL (one of the BOM utilities punched in step 13).

64. Follow BDUPL by the remainder of the system-generation input deck from steps 59 and 60.

65. Ready the 1442.

66. Set sense switch 0 and data switch 15 ON. Press console START.
67. During the execution of BDPIP, refer to "BOM Card Utilities" for BDPIP operating procedures.

68. When execution of BDPIP has been successfully completed, set sense switch 0 and data switch 15 ON and press console START.

69. During execution of BDUPL, refer to "BOM Card Utilities" for BDUPL operating procedures.

70. Unload and save the pack on drive 1 (it is now a copy of the PID pack).

71. Set all sense switches, data switches and program switches OFF; press console STOP, RESET and START.

72. When the following messages are printed on the system printer, set program switch 7 ON and press CONSOLE INTERRUPT:

IBM 1800 DACS MPX/SYS GEN MON 00.000
SEN SW 0 ON ABSOLUTE LOADER
SEN SW 1 ON LOAD BP MONITOR

73. The following messages are printed:

// JOB 00001 00 JAN 00 00.000 HRS
// *
// * AT PAUSE, INSERT YOUR COMPLETED BOM, EXECUTIVE DIRECTOR AND SUBROUTINE
// * EQUATE CARDS, FOLLOWED BY SRFLE END CARDS, BEHIND THEIR RESPECTIVE
// * SRFLE M CONTROL CARDS.
// * PRESS READER AND CONSOLE START TO CONTINUE.
// *
// * BE SURE TO INCLUDE A SRFLE END CARD AT THE END OF EACH
// SET OF EQUATES.
// *
// PAUSE 00 JAN 00 00.000 HRS

74. At the pause, remove all cards from the 1442 stacker, saving the BDUPL and BDPIP card decks and omitting the control cards.

75. At this point you should complete your BOM, Executive Director and Subroutine Library program equate cards (if this has not already been done).

INSERTING EQUATE CARDS

76. Place your completed BOM, Executive Director, and Subroutine Library equates behind their respective SRFLE control cards. The SRFLE control cards are in the system-generation input file in the 1442 card read punch. Be sure that you have included cards containing 9s in the last eight columns behind each set of equates for the library programs (see step 10).

The last set of equates in the library equate deck are for BSCIO and will be placed behind the proper SRFLE control card in a later step. Mark these equates as BSCIO equates and set them aside. Place the remainder of the system-generated input (from steps 46 and 47) into the 1442 hopper.

77. Ready the 1442.

78. Press console START. The following messages are printed, indicating the processing of the equate cards:
ASSEMBLY OF SYSTEM PROGRAMS FOR A SYSTEM WITH ONE OR MORE 1316 DISK PACKS

79. The printout continues:

// JOB 00001 00 JAN 00 00,000 HRS
// THE FOLLOWING CONTROL CARDS WILL DELETE THE ENTIRE COMMUNICATION
// ADAPTER OBJECT LIBRARY. IF THIS IS DESIRED, PRESS START.
// IF IT IS NOT DESIRED SET PROGRAM SWITCH 7 UP AND PRESS CONSOLE INTERRUPT.
// PAUSE 00 JAN 00 00,000 HRS

79a. If your system is to support one or more communications adapters, set program switch 7 ON and press CONSOLE INTERRUPT. Go to step 80 to continue.

79b. If your system does not include communications adapters, press console START and the following programs will be deleted:
CONTINUE
// DMP 00 JAN 00 00.000 HRS
// DELET BSCRT
DMP FUNCTION COMPLETED
// DELET R BSCRT
DMP FUNCTION COMPLETED
// DELET BSCRT
DMP FUNCTION COMPLETED
// DELET BSCCK
DMP FUNCTION COMPLETED
// DELET ZIPCO
DMP FUNCTION COMPLETED
// DELET ZIPCO
CMP FUNCTION COMPLETED
// DELET USHOL
DMP FUNCTION COMPLETED
// DELET EBCUS
DMP FUNCTION COMPLETED
// DELET USPRT
DMP FUNCTION COMPLETED
// DELET USFTP
DMP FUNCTION COMPLETED
// DELET PEFIB
DMP FUNCTION COMPLETED
// DELET PEFIB
DMP FUNCTION COMPLETED
// DELET DPIP
DMP FUNCTION COMPLETED
// DELET RESER
DMP FUNCTION COMPLETED
// DELET DSPRO
DMP FUNCTION COMPLETED

80. The printout continues:

// JOB 00001 00 JAN 00 00.000 HRS
//
// IF THE LIBRARY SUBROUTINES TO SUPPORT THE USE OF 2311's ARE NOT
// NEEDED PRESS START. OTHERWISE, PRESS CONSOLE INTERRUPT WITH
// SENSE SWITCH 7 ON.
//
// PAUSE 00 JAN 00 00.000 HRS

80a. If your system is to support one or more 2311s, set program switch 7 ON and press
CONSOLE INTERRUPT. Go to step 81. If your system is not to support 2311s, go
to step 80b.

80b. If your system does not support 2311s, press console START and the following program:
will be deleted:

CONTINUE
// DMP 00 JAN 00 00.000 HRS
// DELET R DSR
DMP FUNCTION COMPLETED
// DELET R DSR
DMP FUNCTION COMPLETED
// DELET DSAIO
DMP FUNCTION COMPLETED
// DELET DPIP
DMP FUNCTION COMPLETED
// DELET RESER
DMP FUNCTION COMPLETED
// DELET R DSPRO
DMP FUNCTION COMPLETED

// JOB 0000100002 00 JAN 00 00.000 HRS
// THE FOLLOWING CONTROL CARDS WILL DELETE THE 2790 OBJECT AND SOURCE
// LIBRARIES. IF THIS IS DESIRED, PRESS START, OTHERWISE SET SENSE
// SWITCH 7 UP AND PRESS CONSOLE INTERRUPT.
//
// PAUSE 00 JAN 00 00.000 HRS

40 1800 MPX Operating Procedures
81. If your system is to support the 2790 system, set program switch 7 ON and press CONSOLE INTERRUPT. Go to step 82. If your system is not to support the 2790 system, go to step 81a.

81a. If your system does not support the 2790 system, press console START and the following programs will be deleted.

CONTINUE
// DMP 00 JAN 00 00.000 HRS
DELET CLIP
DMP FUNCTION COMPLETED
DELET CLTP
DMP FUNCTION COMPLETED
DELET FILES
DMP FUNCTION COMPLETED
DELET LINKF
DMP FUNCTION COMPLETED
DELET R LOOP1
DMP FUNCTION COMPLETED
DELET R LOOP2
DMP FUNCTION COMPLETED
DELET R WRTLN
DMP FUNCTION COMPLETED
// JOB 00 JAN 00 00.000 HRS.
// DMP 00 JAN 00 00.000 HRS.
DEFINE PAKDK 0
DMP FUNCTION COMPLETED

82. Now assemble the Executive Director.

// JOB 0000100002 00 JAN 00 00.000 HRS.
// X ASSEMBLE EXECUTIVE DIRECTOR, PUNCH THEN STORE.
// ASM EXDIR EXDCD 00 JAN 00 00.000 HRS
OVERFLOW SECTORS 32,0,0
RE-ENTRANT
PUNCH
000 ERROR(S) AND 000 WARNING(S) IN ABOVE ASSEMBLY.

EXDIR SPECL BACK OLEVEL
DMP FUNCTION COMPLETED
STORE EXDIR
EXDIR SPECL BACK OLEVEL
DMP FUNCTION COMPLETED
// X
// NOW ASSEMBLE AND PUNCH BOM
// X
// ASM BOM BOMCD 00 JAN 00 00.000 HRS
OVERFLOW SECTORS 32,0,0
PUNCH
000 ERROR(S) AND 000 WARNING(S) IN ABOVE ASSEMBLY.

BOM
DMP FUNCTION COMPLETED
83. Remove and label the Executive Director and BOM object decks, omitting control cards. Without operator intervention, the system-generation procedure will continue to assemble additional system subroutines.

```
// %
// % THE FOLLOWING CONTROL CARDS ASSEMBLE AND STORE THE SUBROUTINES
// % AFFECTED BY THE JUST-INSERTED LIBRARY EQUATE CARDS
// %
// ASM ADRC ADRCD 00 JAN 00 00.000 HRS
%RE-ENTRANT
000 ERROR(S) AND 000 WARNING(S) IN ABOVE ASSEMBLY.

ADRC
DMP FUNCTION COMPLETED
%STORE 0 ADRC
ADRC
DMP FUNCTION COMPLETED
// ASM DMPS DMPSC 00 JAN 00 00.000 HRS
000 ERROR(S) AND 000 WARNING(S) IN ABOVE ASSEMBLY.

DMPS DUMPS DMPST DMP DUMP DMPHX DMPDC
DMP FUNCTION COMPLETED
%STORE 0 DMPS
DMPS DUMPS DMPST DMP DUMP DMPHX DMPDC
DMP FUNCTION COMPLETED
// ASM CSPAR CSPCD 00 JAN 00 00.000 HRS
%RE-ENTRANT
000 ERROR(S) AND 000 WARNING(S) IN ABOVE ASSEMBLY.

CSPAR CSPLS
DMP FUNCTION COMPLETED
%STORE 0 CSPAR
CSPAR CSPLS
DMP FUNCTION COMPLETED
// ASM LSPCL LSPCD 00 JAN 00 00.000 HRS
%RE-ENTRANT
000 ERROR(S) AND 000 WARNING(S) IN ABOVE ASSEMBLY.

LSPCL LSPPP
DMP FUNCTION COMPLETED
%STORE 0 LSPCL
LSPCL LSPPP
DMP FUNCTION COMPLETED
// ASM MDFIO MDFCD 00 JAN 00 00.000 HRS
%RE-ENTRANT
000 ERROR(S) AND 000 WARNING(S) IN ABOVE ASSEMBLY.

MDFIO MDAF MDAI MDCOM MDF MDFX MDI MDIX MDRED MDWRT
DMP FUNCTION COMPLETED
%STORE 0 MDFIO
MDFIO MDAF MDAI MDCOM MDF MDFX MDI MDIX MDRED MDWRT
DMP FUNCTION COMPLETED
// ASM MFIO MFICD 00 JAN 00 00.000 HRS
%RE-ENTRANT
000 ERROR(S) AND 000 WARNING(S) IN ABOVE ASSEMBLY.

MFIO MRED MWRT MCOMP MIOAF MIOAI MIOFX MIOIX MIOF MIOI
DMP FUNCTION COMPLETED
%STORE 0 MFIO
MFIO MRED MWRT MCOMP MIOAF MIOAI MIOFX MIOIX MIOF MIOI
DMP FUNCTION COMPLETED
```

40.2 1800 MPX Operating Procedures
84. At the completion of step 83, the following is printed:

```plaintext
// JOB 0000100002 00 JAN 00 00.000 HRS
// X
// X THE FOLLOWING FUNCTIONS ARE CONCERNED WITH THE ASSEMBLY OF THE
// X COMMUNICATIONS IOCR (BSCIO). IF THIS SYSTEM DOES NOT SUPPORT
// X COMMUNICATIONS PRESS CONSOLE INTERRUPT WITH SENSE SWITCH 7 UP,
// X OTHERWISE PRESS START.
// X
// PAUSE 00 JAN 00 00.000 HRS
```

85. If your system is to support no communications adapters, set program switch 7 ON
and press CONSOLE INTERRUPT. N11 error messages will be printed and are to be
ignored. Go to step 86 to continue.

85a. If your system is to support one or more communications adapters, press START and
the following messages are printed:

```plaintext
CONTINUE
//
// AT PAUSE PLACE BSCIO SYSGEN EQUATES AFTER "SRFLE CONTROL CARD
// INCLUDE A SRFLE END CARD AT THE END OF THE EOU'S
// PRESS READER START AND CONSOLE START TO CONTINUE
// 
// PAUSE 00 JAN 00 00.000 HRS
```

85b. Insert the BSCIO equate cards behind the *SRFLE card in the 1442. Ready the 1442
and press console START. The following is printed:

```plaintext
CONTINUE
// DMP 00 JAN 00 00.000 HRS
"SRFLE M BSCCD
000005 CARDS READ
000011 SECTORS AVAILABLE
DMP FUNCTION COMPLETED
//
// ASSEMBLE AND PUNCH BSCIO
//
// ASM BSCIO BSCCD 00 JAN 00 00.000 HRS
"RE-ENTRANT
"PUNCH
000 ERROR(S) AND 000 WARNING(S) IN ABOVE ASSEMBLY.
BSCIO
DMP FUNCTION COMPLETED
"STORE 0 BSCIO
BSCIO
DMP FUNCTION COMPLETED
```
85c. Remove and label the BSCIO object deck, omitting the control cards.

86. The printout continues:

```
// JOB 0000100002 00 JAN 00 00.000 HRS
//
// THE FOLLOWING SOURCE FILES AND UTILITIES ARE DELETED
// DELETE-- EQUBM, EQUEX, EQULB, GENCD, CSPCD, LSPCD, MDFCD, MFID, EXDCD
// BLIST, BDWAP, SYSLD, BDUPL, BDPAT, BDPIP, BDCL, ADRC, DMPSC, SAMP1
// BOMCD, BSCCD
// IF ALL PRECEDING OPERATIONS COMPLETED SUCCESSFULLY,
// PRESS CONSOLE START TO CONTINUE.
//
// PAUSE PRESS START TO CONTINUE 00 JAN 00 00.000 HRS
```

If your system generation has not proceeded successfully to this point, don't continue. Correct the error and go back to the appropriate step to continue the generation process.

If your system generation has proceeded successfully to this point, press console START. This will cause the deletion of system-generation source files on the disk and bring you to the end of phase 2.
2790 INSTALLATION

If you are installing the 2790 system, it is necessary to store to disk drive 0 the On Line 2790 Area Station Exerciser/Customer Engineer Diagnostic Program. The object card deck for this program can be obtained from your customer engineer and stored in core image format using the following job steps:

```
11-20  21-30  31-40  41-50  51-60
12314617181920112314617181920112314617181920112314617181920112314617181920
```

These cards are included with the object card deck.

This completes phase 2.

ASSEMBLY OF SYSTEM PROGRAMS FOR A SYSTEM WITH 2315 DISK CARTRIDGES

87. The following is printed:

```
// JOB  00001  00 JAN 00 00,000 HRS
// * THE FOLLOWING CONTROL CARDS WILL DELETE THE ENTIRE COMMUNICATION
// * ADAPTER OBJECT LIBRARY. IF THIS IS DESIRED, PRESS START.
// * IF IT IS NOT DESIRED SET PROGRAM SWITCH 7 UP AND PRESS CONSOLE INTERRUPT.
// * PAUSE  00 JAN 00 00,000 HRS
```

88. If your system is to support one or more communications adapters, set program switch 7 ON and press CONSOLE INTERRUPT. Go to step 89 to continue.
88a. If your system does not include communications adapters, press console START and the following programs will be deleted:

```
CONTINUE
// DMP	 00 JAN 00 00.000 HRS
DELET	 BSCRT
DMP FUNCTION COMPLETED
DELET	 BSCTR
DMP FUNCTION COMPLETED
DELET	 BSCTR
DMP FUNCTION COMPLETED
DELET	 BSCCK
DMP FUNCTION COMPLETED
DELET	 ZIPCO
DMP FUNCTION COMPLETED
DELET	 ZIPCO
DMP FUNCTION COMPLETED
DELET	 USHOL
DMP FUNCTION COMPLETED
DELET	 EBCUS
DMP FUNCTION COMPLETED
DELET	 USPRT
DMP FUNCTION COMPLETED
DELET	 USTYP
DMP FUNCTION COMPLETED
DELET	 PEFIB
DMP FUNCTION COMPLETED
DELET	 PEFIB
DMP FUNCTION COMPLETED
DELET	 FLPBE
DMP FUNCTION COMPLETED
DELET	 FLPBE
DMP FUNCTION COMPLETED
```

89. The printout continues:

```
// JOB	 00001	 00 JAN 00 00.000 HRS
// 
// IF THE LIBRARY SUBROUTINES TO SUPPORT THE USE OF 2311'S ARE NOT
// NEEDED PRESS START. OTHERWISE, PRESS CONSOLE INTERRUPT WITH
// SENSE SWITCH 7 ON.
// 
// PAUSE	 00 JAN 00 00.000 HRS
```

89a. If your system is to support one or more 2311s, set program switch 7 ON and press CONSOLE INTERRUPT. Go to step 90.

89b. If your system does not support 2311s, press console START and the following programs will be deleted:

```
CONTINUE
// DMP	 00 JAN 00 00.000 HRS
DELET	 DSOR
DMP FUNCTION COMPLETED
DELET	 DSCR
DMP FUNCTION COMPLETED
DELET	 MADAO
DMP FUNCTION COMPLETED
DELET	 DPIP
DMP FUNCTION COMPLETED
DELET	 0 DSPO
DMP FUNCTION COMPLETED
```

```
// JOB	 00001	 00 JAN 00 00.000 HRS
// 
// THE FOLLOWING CONTROL CARDS WILL DELETE THE 2790 OBJECT LIBRARY.
// IF THIS IS DESIRED, PRESS START, OTHERWISE SET SENSE SWITCH 7 UP
// AND PRESS CONSOLE INTERRUPT.
// 
// PAUSE	 00 JAN 00 00.000 HRS
```

42.2 1800 MPX Operating Procedures
90. If your system is to support the 2790 system, set program switch 7 ON and press CONSOLE INTERRUPT. Go to step 91.

90a. If your system does not support the 2790 system, press console START and the following programs will be deleted:

```
CONTINUE
// DMP 00 JAN 00 00.000 HRS
#DELETE CLIP
DMP FUNCTION COMPLETED
#DELETE CLTP
DMP FUNCTION COMPLETED
#DELETE FILES
DMP FUNCTION COMPLETED
#DELETE LINK1
DMP FUNCTION COMPLETED
#DELETE R LOOP1
DMP FUNCTION COMPLETED
#DELETE R LOOP2
DMP FUNCTION COMPLETED
#DELETE WRTLN
DMP FUNCTION COMPLETED
```

// JOB 00 JAN 00 00.000 HRS.
// DMP 00 JAN 00 00.000 HRS.
#DEFINE PAKD 0
DMP FUNCTION COMPLETED

91. The Executive Director and BOM are now assembled. These operations may take as long as two hours, depending on the main-storage size of your system. At the completion of the previous step, the following messages are printed, indicating that the Executive Director and BOM have been assembled.

```
// JOB 000000111112 00 JAN 00 00.000 HRS
// # ASSEMBLE EXECUTIVE DIRECTOR, PUNCH THEN STORE.
// ASM EXDIR EXDCD 00 JAN 00 00.000 HRS
#OVERFLOW SECTORS 32,0,0
#RE-ENTRANT
#PUNCH

000 ERROR(S) AND 000 WARNING(S) IN ABOVE ASSEMBLY.
EXDIR SPECL BACK QLEVEL
DMP FUNCTION COMPLETED
#STORE EXDIR
EXDIR SPECL BACK OLEVEL
DMP FUNCTION COMPLETED
#STORE EXDIR
DMP FUNCTION COMPLETED
//
// # NOW ASSEMBLE AND PUNCH BOM
// # IF YOU GET A DLL NEED BLANK CARDS ERROR MESSAGE
// # NPRO THE CARDS IN THE CARD READER AND INSERT BLANK CARDS.
//
// ASM BOM BOM 00 JAN 00 00.000 HRS
#OVERFLOW SECTORS 32,0,0
#PUNCH

000 ERROR(S) AND 000 WARNING(S) IN ABOVE ASSEMBLY.
```

BOM
DMP FUNCTION COMPLETED
92. Remove and label the Executive Director and BOM object decks, omitting control cards. Without operator intervention, the system-generation procedure will continue to assemble additional system subroutines.

---

42.4 1800 MPX Operating Procedures
93. At the completion of step 92, the following is printed:

// JOB  00001 00 JAN 00 00.000 HRS
// JOB  THE FOLLOWING FUNCTIONS ARE CONCERNED WITH THE ASSEMBLY OF THE COMM-
// JOB  UNICATIONS ADAPTER IOCA (BSCIO) AND BUILDING OF THE 2790 FILE ON DR 1.
// JOB  IF THIS SYSTEM DOES NOT SUPPORT COMMUNICATIONS ADAPTER AND 2790,
// JOB  PRESS CONSOLE INTERRUPT WITH SENSE SWITCH 7 UP, OTHERWISE PRESS START.
// JOB  IF CONSOLE INTERRUPT IS USED IGNORE ALL ERRORS OF THE FORM
// JOB  N11 LABEL ERR DR N
// JOB  PAUSE 00 JAN 00 00.000 HRS

93a. If your system does not support the communications adapter or the 2790 system, set
program switch 7 ON and press CONSOLE INTERRUPT. If this option is taken, N11
LABEL ERR DR N will be printed until the next JOB 00001 card is encountered. Go
to step 102.

94. If your system is to support one or more communications adapters or the 2790 system,
press START and the following messages will be printed:

CONTINUE
// JOB  AT PAUSE PRESS STOP RESET AND START
// JOB  MOUNT PID PACK 3 ON DRIVE 0 AND A SCRATCH PACK ON DRIVE 1.
// JOB  WHEN MPP/BOM MESSAGES ARE PRINTED, NPRO READER, PLACE BDWAP DECK IN
// JOB  READY READER, TURN ON SS 0, DATA SWITCH 15, AND PRESS CONSOLE START.
// JOB  LEAVE DATA SWITCH 15 ON FOR ONE TRY. PRESS CONSOLE START.
// JOB  LEAVE DATA SWITCH 15 ON FOR DRIVE 1. PRESS CONSOLE START.
// JOB  AT COMPLETION OF BDWAP, TURN OFF CONSOLE SWITCHES, PRESS STOP, RESET
// JOB  AND START. THEN SET PROGRAM SW 7, AND PRESS CONSOLE INTERRUPT TO CONTINUE
// JOB  PAUSE 00 JAN 00 00.000 HRS

// JOB  00003 00 JAN 00 00.000 HRS
// JOB  DMP 00 JAN 00 00.000 HRS
// JOB  DMP FUNCTION COMPLETED

// JOB  000031113 00 JAN 00 00.000 HRS
// JOB  DMP 00 JAN 00 00.000 HRS
// JOB  DEFINE CONFG C1 0150
// JOB  DMP FUNCTION COMPLETED

// JOB  000031113 00 JAN 00 00.000 HRS
// JOB  THE FOLLOWING FUNCTIONS SETUP THE 2790 LIBRARY ON DRIVE 1.
// JOB  IF THIS SYSTEM DOES NOT SUPPORT 2790, PRESS CONSOLE INTERRUPT WITH
// JOB  SENSE SWITCH 7 UP, OTHERWISE PRESS START.
// JOB  PAUSE 00 JAN 00 00.000 HRS

95. If your system does not support the 2790 system, set sense switch 7 ON and press
CONSOLE INTERRUPT. Go to step 97.
96. If your system is to support the 2790 system, the A2790 file will be defined on drive 1.
Press START to continue. The following is printed:

CONTINUE
// DMP 00 JAN 00 00.000 HRS
DFILE 1 A2790 950
WILL RESERVE AT SCTR ADDR 1190
DMP FUNCTION COMPLETED
MACRO UPDATE
SELECT
BUILD 'A2790'
JOIN 'X2790'

97. If your system does not support any communications adapters, set sense switch 7 ON
and press CONSOLE INTERRUPT. Go to step 100.

98. If your system supports any communications adapters, the BSCIO file will be set up
on drive 1. Press START to continue. The following is printed:

CONTINUE
// DMP 00 JAN 00 00.000 HRS
DFILE 1 BSCIO 250
WILL RESERVE AT SCTR ADDR 1546
DMP FUNCTION COMPLETED
//
// AT PAUSE PLACE BSCIO SYSGEN EQUATES AFTER *SRFLE CARD, INCLUDE A SRFLE
// END CARD AT THE END OF THE EQUATES. PRESS READER AND CONSOLE START.
//
// PAUSE 00 JAN 00 00.000 HRS

99. Insert the BSCIO equate cards behind the *SRFLE card in the 1442. Ready the 1442
and press console START. The following is printed:

CONTINUE
// DMP 00 JAN 00 00.000 HRS
MSRFLE M BSCCD BSCIO
003293 CARDS ON FILE
000239 SECTORS USED
000011 SECTORS AVAILABLE
000005 CARDS READ
DMP FUNCTION COMPLETED
// JOB 0000311113 00 JAN 00 00.000 HRS
// REMOVE PID PACK 3 FROM DRIVE 0 AND SAVE.
// MOUNT COPY OF PID PACK 1 ON DRIVE 0.
//
// WHEN THE DRIVE IS READY PRESS STOP, RESET AND START.
// THEN WITH SENSE SWITCH 1 UP, PRESS START.
//
// PAUSE 00 JAN 00 00.000 HRS
100. When drive 0 is READY, press STOP, RESET, and START. Set sense switch 1 ON and press START. The following is printed:

```
// JOB 0000111113 00 JAN 00 00.000 HRS
// ** THE FOLLOWING FUNCTIONS ASSEMBLE AND STORE BSCIO.
// ** IF THIS SYSTEM DOES NOT SUPPORT COMM ADAPTER PRESS CONSOLE INTERRUPT
// ** WITH SENSE SWITCH 7 UP, OTHERWISE PRESS START.
// **
// ** PAUSE 00 JAN 00 00.000 HRS
```

101. If your system supports any communications adapters, press START to assemble and store BSCIO.

101a. If your system does not support communications adapters, set sense switch 7 ON, press CONSOLE INTERRUPT, and go to step 102.

CONTINUE

```
// ASM BSCIO BSCIO 00 JAN 00 00.000 HRS
#PRE-ENTRANT
#PUNCH

000 ERROR(S) AND 000 WARNING(S) IN ABOVE ASSEMBLY.
```

BSCIO

DMP FUNCTION COMPLETED

```
#STORE 0 BSCIO
BSCIO
DMP FUNCTION COMPLETED
```

102. The printout continues:

```
// JOB 00001 00 JAN 00 00.000 HRS
// **
// ** THE FOLLOWING SOURCE FILES AND UTILITIES ARE DELETED
// ** DELETE-- EQUBM, EQUEX, EQULB, GENCD, CSPCD, LSPCD, MDPCD, MFICD, EXDCD
// ** BLIST, BDWAP, SYSLD, BDUPL, BDIP, BDPIP, BDICL, ADRC, DMPSC, #AMP1
// ** IF ALL PRECEDING OPERATIONS COMPLETED SUCCESSFULLY,
// ** PRESS CONSOLE START TO CONTINUE.
// **
// ** PAUSE PRESS START TO CONTINUE 00 JAN 00 00.000 HRS
```
If your system generation has not proceeded successfully to this point, don’t continue. Correct the error and go back to the appropriate step to continue the generation process.

If your system generation has proceeded successfully to this point, press console START. This will cause the deletion of system-generation source files on the disk and bring you to the end of phase 2.

CONTINUE

// DMP 00 JAN 00 00,000 HRS
*DELET D EDUBM
DMP FUNCTION COMPLETED
*DELET D EQUEX
DMP FUNCTION COMPLETED
*DELET D EQLIB
DMP FUNCTION COMPLETED
*DELET D GENCD
DMP FUNCTION COMPLETED
*DELET D CSPCD
DMP FUNCTION COMPLETED
*DELET D LSPCD
DMP FUNCTION COMPLETED
*DELET D MDFIO SOURCE
DMP FUNCTION COMPLETED
*DELET D MFICO
DMP FUNCTION COMPLETED
*DELET D EXDCD
DMP FUNCTION COMPLETED
*DELET D BLCX
DMP FUNCTION COMPLETED
*DELET D BDUMP
DMP FUNCTION COMPLETED
*DELET D BDUPL
DMP FUNCTION COMPLETED
*DELET D BDPAT
DMP FUNCTION COMPLETED
*DELET D BDIP
DMP FUNCTION COMPLETED
*DELET D BUCRL
DMP FUNCTION COMPLETED
*DELET D BDRD
DMP FUNCTION COMPLETED
*DELET D ADRCM
DMP FUNCTION COMPLETED
*DELET D DMPSC
DMP FUNCTION COMPLETED
*DELET D SAMPL
DMP FUNCTION COMPLETED

// JOB 00 JAN 00 00,000 HRS
// DMP 00 JAN 00 00,000 HRS
*DEFINE PAKK 0
DMP FUNCTION COMPLETED

// JOB 00 JAN 00 00,000 HRS
// * AT THIS POINT REFER TO OP MANUAL FOR FURTHER INSTRUCTIONS.
// END 2790 USERS SHOULD REFER TO 2790 SRL FOR BUILDING CORELOADS.

This completes phase 2.

42.8 1800 MPX Operating Procedures
(Pages 42.9 and 42.10 deleted)
Information formerly on this page has been moved to page 49.
PHASE 3 OPERATIONS

LOADING BOM

1. Remove any cards that may be in the 1442.
2. Clear main storage to zeros (refer to "I/O Device and Console Operations").
3. Place the 4-card BOM high core loader (received from PID) followed by the BOM deck (punched during phase 2, step 82) into the 1442 hopper.
4. Ready the 1442 and the 2311 drives.
5. Press PROGRAM LOAD on the console. The following messages are printed:

```
IBM 1800 MPX/BOM XX, XXX  00 JAN 00
SEN SW 0 ON ABSOLUTE LOADER
SEN SW 1 ON LOAD BP MONITOR
SEN SW 2 ON SET CLOCK VIA DATA SWS
SEN SW 3 ON SET DATE VIA DATA SWS
```

LOADING THE ASSIGNMENT CARDS

6. Punch your assignment cards (see Appendix A) and the System Loader control cards shown below.
7. Place the System Loader deck (punched during phase 2), the system loader control cards and your completed assignment cards into the 1442 hopper in the following order:

```
<table>
<thead>
<tr>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
</tr>
</thead>
<tbody>
<tr>
<td>12356789012356789012356789012356789</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

8. Set sense switch 0 ON, data switch 15 ON and press console START. The I/O device and interrupt assignments you have selected will be printed by the System Loader.

To insure that valid FORTRAN logical unit numbers are inserted into the IOU subroutine, IOU must be on drive 0.

This completes phase 3.
PHASE 4 OPERATIONS

In this phase you will define the number of drives in your system, label those drives and define your main-storage layout. The sequences of control cards used to carry out phase 4 operations are illustrated in Figures 2 and 3. Note that each job for each drive requires a separate // JOB and // DMP card; the // JOB statements don't contain drive labels, because the drives haven't been initialized yet. Refer to IBM 1800 Multiprogramming Executive Operating System Programmer's Guide, Order Number GC26-3720, for details on how to complete and use each of these control statements.

OPERATING PROCEDURES

1. If applicable, place the scratch packs to be labeled on their respective drives and ready those drives.

2. Place the deck of control cards to define and label disks and to define your main-storage layout in the 1442 hopper in the sequence illustrated in Figures 2 and 3.

3. Press START on the 1442. This will cause the cards to be read and processed. This completes phase 4.
See the MPX Programmer’s Guide for a detailed description of data to be entered.

XX equals 16, 24, 32, 40, 48, 56, 64, depending on the size of main storage.

Figure 3. Define Main-Storage and Disk Layout

PHASE 5 OPERATIONS

Phase 4 operations finished with the Batch-Processing Monitor Supervisor in control. Batch-processing operations continue, but are divergent at this point, with one procedure for generating a batch-processing system and another for generating a real-time system.

BATCH-PROCESSING SYSTEM

To complete the generation of your batch-processing system, you build BOM, using the control statements and BOM object deck illustrated in Figure 4.

1. Place the control cards to build BOM and the BOM object deck in the 1442 hopper in the sequence illustrated in Figure 4.

2. Press the START on the 1442. This will cause the cards to read and processed.
This completes the system-generation process for a batch-processing system. To start your system you will use the seven cold start loader cards and a cold start name card (Figure 5). For details on cold starting a system, see "Cold Start."

Before you begin batch processing, you may want to use the CE Coreload (described later under "CE Coreload Programs") to initialize system error counters.

REAL-TIME SYSTEM

To complete the generation of your real-time system, you assemble or compile and store any subroutines to be included in the Executive, build your Executive, and assemble or compile and build your process coreloads. The sequences of control statements necessary to complete phase 5 are illustrated in Figures 6 and 7. Refer to IBM 1800 Multiprogramming Executive Operating System Programmer's Guide, Order Number GC26-3720, for details on how to complete and use each of these control statements.

1. Place the completed control statements and source decks in the 1442 hopper in the order illustrated in Figure 6.
2. Place the completed control statements and BOM object deck in the 1442 hopper in the order illustrated in Figure 7.

3. Press START on the 1442. This will cause the cards to be read and processed.

This completes the system-generation process for a real-time system. Remaining is the assembling or compiling and building of your process coreloads. At this point you may wish to start your real-time system or you may continue under the Batch-Processing Monitor. If you wish to continue under the Batch-Processing Monitor, skip to step 4.

To start your real-time system you must perform a cold start to the Batch-Processing Monitor (see "Cold Start"). When the cold start is complete, continue with step 4.

4. Complete the control statements necessary to assemble or compile and build your process coreloads as illustrated in Figure 8.

5. Place the completed control statements and source decks in the 1442 hopper.

6. Press START on the 1442. This will cause the cards to be read and processed.
2790 INSTALLATION

If you are installing the 2790 system, it is necessary to store to disk drive 0 the On Line 2790 Area Station Exerciser/Customer Engineer Diagnostic Program. The object card deck for this program can be obtained from your customer engineer and stored in core image format using the following job steps:

These cards are included with the object card deck.

This completes phase 5. To start your real-time system you must cold start to either the Batch-Processing Monitor or a process coreload (see "Cold Start").

Before you begin real-time processing, you might want to use the CE Coreload (described later under "CE Coreload Programs") to initialize system error counters.
REBUILDING THE SYSTEM EXECUTIVE

SYSTEM EXECUTIVE REBUILD CONDITIONS

While relocatable programs can be deleted and replaced on-line by the Batch Processing Monitor, it is not possible to modify any features of the System Executive on-line. Changes in the Executive area (including Executive interrupt routines) thus require an off-line Executive rebuild.

The MPX System Executive may be rebuilt at any time by following the detailed operating procedures specified for an initial Executive build, which is discussed elsewhere in the MPX Operating Procedures manual.

Since INSKEI COMMON is not open-ended, the user may face the difficulty of adding to it after it is defined. It is recommended that an area be reserved in INSKEI COMMON to allow for programming contingencies. See Core Load Rebuild Conditions.

Figure 8. Compile and/or Assemble User-Written Programs and Build Process Coreloads for Execution
The Executive Branch Table (EBT) has already been described. This table enables the user to rebuild the System Executive when modifying subroutines, changing the logic flow or adding patches to the Executive Director and BOM. After such rebuilding, addresses in core loads will still reference a fixed address in the EBT. An ability to shift the entry points of subroutines within the Executive is thus available without the necessity of rebuilding the referencing core loads.

If, however, the entry points within these tables no longer pointed to the same subroutine, all core loads must be rebuilt (see Core Load Rebuild Conditions).

When the Executive is initially built, the entry points to the in-core-with-Executive (ICI) routines that are to be serviced by a level and bit (LLBB) designation are placed in the Interrupt Core Load Table (ICLT), and all other entry spaces are zeroed out. Subsequently, when a core load is built and queued to an LLBB designation (out-of-core interrupt servicing), an entry is made into the ICLT in the corresponding LLBB position which indicates the location on disk where the out-of-core interrupt servicing core load resides.

However, if it is desirable to rebuild the Executive, the information reflected in the ICLT for out-of-core interrupts can be preserved. This is accomplished by rebuilding the Executive with the SAVE ICLT OPTION (by placing an S in column 20 of the //XEQ . EXEC control card). This effectively copies the word count and sector address for each out-of-core interrupt from the old ICLT into a corresponding LLBB position of the new (fresh) ICLT, providing a new ICI was not included in the new Executive for this LLBB position.

CORE LOAD REBUILD CONDITIONS

In an Executive rebuild, it may not be necessary to rebuild those core loads built under the previous (that is, old) Executive if the following conditions are met:

1. No previously included CALL, LIBF, ICI, or ISS type subroutines may be removed from the Executive. However, additional CALL or ISS type subroutines may be added as dictated under condition 5.
2. All previously included subroutines may be reassembled or recompiled in preparation for an Executive rebuild, provided the names and the number of entry points for these subroutines are not modified between inclusions.
3. No additional LIBF, ICI, or ISS type subroutines (types 3, 4, and 5 respectively) may be included in the new Executive.
   CALL or ISS type subroutines (types 4 and 6) may, however, be added provided that Patch Area in the Executive is large enough to contain these additions.
4. If BOM or the Executive Director is reassembled, the number of interrupt levels used, the length of INSKEL COMMON, and the length of the System Executive Area should not be altered.
   Note also that a change in the size of INSKEL COMMON implies a reassembly of BOM, and a resulting alteration in the high-address origin of INSKEL COMMON. Any new core load built at this time will therefore be incompatible with previously-built core loads using INSKEL COMMON. In addition, if INSKEL COMMON were increased in size, old core loads would remain compatible with each other in their use of this common area. However, if the size of INSKEL COMMON were decreased, previously built core loads using INSKEL COMMON would be invalidated.
5. If there are to be additions to user-included subroutines in the Executive as permitted under condition 3, the order of *INCLD control cards in the Executive rebuild process is critical.

   In the rebuild operation, three sets of *INCLD control cards A, B, and C are required in that order.
   
   Set A must correspond to the *INCLD control cards used for the previous Executive build.
   
   Set B must be made up by the user and contain the names of the Builder-included subroutines pertinent to the previous Executive build. These names appear in the Executive Core Map (of the previous Executive build), and must be specified on *INCLD control cards in this exact order.
   
   Set C must contain the names of the new user-included subroutines.
   
   See Example of Executive Build and Rebuild.

6. Following the rebuild process, a visual comparison of the previous Executive and the new Executive core maps must show identical entry points for those LIBF and CALL map entries common to both executives.

EXAMPLE OF INITIAL EXECUTIVE BUILD AND EXECUTIVE REBUILD

Program Listing 16 illustrates the general sequence of control cards, the Executive Core Map, the Interrupt Core Load Table (ICLT), and the Executive Tables for a typical initial executive build and executive rebuild. A separate interpretation of the Executive Core Map and the ICLT is given at the end of this section.

In the initial Executive:

FADD is an IBM MPX Library subroutine included in the Executive for the purpose of being shared by various core areas.

CALLA and CALLB are two subroutines included in the Executive for the purpose of servicing process interrupts on level 10 bit 2 and level 10 bit 10 respectively.

Note that these two assignments are printed in the ICLT map.

CALLC is a special user-written CALL-type subroutine included in the Executive for the purpose of being shared by various core load areas.

In the rebuilt Executive:

No major modifications are implemented. Neither the Executive I/O nor INSkel COMMON are altered.

The interrupt servicing subroutines CALLA and CALLB have been reassigned to service level 14 bit 1 and level 2 bit 2 respectively. These pertain to set A *INCLD control cards.

Set B *INCLD control cards contains those Builder-included subroutines which were implicitly added in the previous build.

Set C *INCLD control cards gives the additional subroutines included by the user. These are:

ISSA - a user-included subroutine for the express purpose of servicing an RPQ device interrupt.

CALLD - a user-included CALL-type subroutine to be used by the various core load areas.

You should be aware that in rebuilding the Executive, the set A *INCLD control cards must be in exact order-correspondence with the previous Executive build, so that these routines will be loaded in the same order-sequence, and entry points in the Executive Branch Table and Executive Transfer Vector remain valid.
Program Listing No. 16

Initial Executive Build

// JOB A 00,000 HRS

// XEQ .EXEC L FXO
HINCLUDFADD
HINCLUDCALLA/1002
HINCLUDCALLB/1010
HINCLUDCALLC
HCCEND

MPX, BUILD .EXEC
EXEC CORE MAP
TYPE NAME ARG1 ARG2
CALL BULKN 0644 007C
CALL TYPEN 0AA4 0089
CALL WRTYN 0AA4 0089
CALL LINK 380C 008E
CALL EXIT 3508 0086
CALL PRMTN 1143 008A
CALL RSAVE 1445 00C7
CALL ABOIT 17FF 0039
CALL EACRL 17CE 00C6
CALL EACPT 1692 0099
CALL GETQ 1797 00FC
CALL PUTQ 1739 00FB
CALL IOTST 14D1 0062
CALL IOSET 148B 0063
CALL IOSAV 138C 0075
CALL QZEXT 1414 0098
CALL QZSAV 13F5 009A
CALL TVEXT 1303 00AD
CALL TVSAV 13B3 00AC
COMM 01AE 0175
ORG EXDIR 26D6
CLNT EXEC 3E3E
ICICALLA 3C44 0A02
ICICALLB 3C46 0A0A
LIBF FADD 3C66 3E5E R
CALL CALLC 3C65 3E76
CALL SPECL 3C70 3E75 R
CALL QLEVL 3B7F 3E74 R
LIBF FSBR 3C4B 3E61 R
LIBF FSBRX 3C4D 3E64 R
LIBF FSUB 3C5C 3E67 R
LIBF FADDX 3C62 3E6A R
LIBF FSUBX 3C57 3E6D R
LIBF FARC 3C68 3E70 R
PITCH 3D1C 3E3B

ICL TABLE MAP
LLBB WC/EP 3A ICLT

0A02 3C44 2B66
0A0A 3C46 2B66

Executive Tables

<table>
<thead>
<tr>
<th>CLNT</th>
<th>ICI</th>
<th>LIBF</th>
</tr>
</thead>
<tbody>
<tr>
<td>0A02</td>
<td>3C44</td>
<td>2B66</td>
</tr>
<tr>
<td>0A0A</td>
<td>3C46</td>
<td>2B66</td>
</tr>
</tbody>
</table>

Cold Starting a Real-Time System 50.3
Executive Rebuild

// JOB	 A 00.099 HRS

// XEQ .EXEC L FXO
\#INCLDFADD
\#INCLDCALLA/1401
\#INCLDCALLB/0202
\#INCLDCALLC
\#INCLDCALLD
\#CCEND

Set A
\#INCLDCALLA/1401
\#INCLDCALLB/0202
\#INCLDCALLC
\#INCLDCALLD

Set B
\#INCLDCALLA/1401
\#INCLDCALLB/0202
\#INCLDCALLC

Set C
\#INCLDCALLA/1401
\#INCLDCALLB/0202
\#INCLDCALLC

MPX, BUILD .EXEC
EXEC CORE MAP
TYPE NAME ARG1 ARG2
CALL BULKN 0644 007C
CALL TYPEN OAA4 0089
CALL WRTYN OAA4 0089
CALL LINK 380C 008E
CALL EXIT 350B 0086
CALL PRNTN 1143 008A
CALL RSAVE 14F5 00C7
CALL ABORT 17FF 0039
CALL EACRL 17CE 0066
CALL EACPT 1C92 0099
CALL GETQ 1797 00FC
CALL PUTQ 1739 00FB
CALL IOTST 14D1 0062
CALL IOSET 14B8 0063
CALL IOEXT 1359 0076
CALL IOSAV 138C 0075
CALL QZEXT 1359 009B
CALL QZSAV 138E 00AC
CALL TVEXT 138E 00AE
CALL TVSAV
CALL IOTST 14D1 0062
CALL IOSET 14B8 0063
CALL IOEXT 1359 0076
CALL IOSAV 138C 0075
CALL QZEXT 1359 009B
CALL QZSAV 138E 00AC
CALL TVEXT 138E 00AE
CALL TVSAV
COMM 01AE 0275
ORG EXDIR 26D6
CLNT .EXEC
3E3C
ICI CALLA 3C44 0E01
ICI CALLB 3C46 0202
LIBF FADD 3C68 3E5C R
LIBF CALLC 3C66 3E76 R
LIBF SPECL 3E42 3E75 R
LIBF BACK 3E47 3E74 R
LIBF QLEVEL 3E49 3E73 R
LIBF FSBR 3C48 3E5F R
LIBF FSBRX 3C4D 3E62 R
LIBF FSUB 3C5C 3E65 R
LIBF FADDX 3C62 3E68 R
LIBF FSUX 3C57 3E6B R
LIBF FARC 3C68 3E6E R
LIBF FARCX 3C6D 3E70 R
LIBF FSBR 3C48 3E5F R
LIBF FSBRX 3C4D 3E62 R
LIBF FSUB 3C5C 3E65 R
LIBF FADDX 3C62 3E68 R
LIBF FSUX 3C57 3E6B R
LIBF FARC 3C68 3E6E R
LIBF FARCX 3C6D 3E70 R
LIBF CALLD 3D24 3E71
PTCH 3E26 3E39

ICL TABLE MAP
LL99 WC/EP SA ICLT
0202 3C46 282E
0E01 3C46 2954

Executive Tables
3E00 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
3E30 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
3E40 0480 0089 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
3E50 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
3E60 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
3E70 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
3E80 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000

50.4 1800 MPX Operating Procedures
INTERPRETATION OF THE EXECUTIVE CORE MAP AND THE INTERRUPT CORE LOAD
TABLE (ICLT) MAP

EXECUTIVE CORE MAP

The Executive Build function always prints a map of the assembled Executive in the following
format:

<table>
<thead>
<tr>
<th>EXEC</th>
<th>CORE</th>
<th>MAP</th>
<th>(Page Heading)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TYPE</td>
<td>NAME</td>
<td>ARG1</td>
<td>ARG2</td>
</tr>
</tbody>
</table>

Type indicates the map entry type (for example, LIBF, CALL, and CLNT). Up to five
alphameric characters are permitted under NAME to describe a subroutine, control program,
etc. (for example, BULKN, EXDIR).

ARG1 and ARG2 may contain either a four-digit hexadecimal number or a blank field.

Core Load Name Table (CLNT)

| CLNT | NNNNN | XXXX | YYYY |

The word count and disk address of the core load named NNNNN, which is referenced
within the Executive, are assigned to locations YYYY and YYYY+1 of the Executive Core
Load Name Table. The XXXX field is blank for all CLNT entries.

Executive Director Origin (ORG)

| ORG | EXDIR | XXXX | YYYY |

XXXX is the hexadecimal address of the Executive Director: More precisely, it is the
beginning of the ICLT pointer block. YYYY is always blank.

In-Core-With-Executive Interrupt Routines (ICI)

| ICI | NNNNN | XXXX | LLBB |

The entry point to the in-core-with-Executive interrupt servicing routine named NNNNN is
at absolute location XXXX. LL designates the interrupt level (ILSW) while BB designates
the interrupt level (ILSW) while BB designates the bit position within the PISW for that
associated level.

Library Function Subroutines (LIBF)

| LIBF | NNNNN | XXXX | YYYY |

The LIBF-type subroutine entry point named NNNNN is at absolute location XXXX of the
Executive. The corresponding three-word transfer vector entry point will be at the location
YYYY in the ETV Table in the System Executive.

Call-type Subroutines (CALL)

| CALL | NNNNN | XXXX | YYYY |

The Call-type subroutine entry point named NNNNN is at absolute location XXXX of the
Executive. The indirect entry point is at location YYYY of the System Executive Branch
Table (EBT).
The low core storage boundary of INSKEL COMMON is at absolute location XXXX of the Executive. The high boundary is at location YYYY.

Patch Area (PTCH)

The Patch Area (that is, unused core locations) extends from XXXX, the absolute location of the Executive, through location YYYY.

INTERRUPT CORE LOAD TABLE (ICLT) MAP

The ICL Table map is printed to reveal any interrupt assignments made in the Executive ICLT. Its format is as follows:

ICL TABLE MAP
LLBB WC/EP SA ICLT (Column Heading)

The interrupt level and bit assignments are indicated by a four-digit hexadecimal number under LLBB. The two high-order digits contain the level; the two low-order digits represent the bit assignment.

If the entry is an in-core-with-Executive routine, the WC/EP column will contain the hexadecimal entry point to this routine. This address corresponds to the ARG1 address (that is, XXXX) of ICI entries. The SA field will be blank. The ICL Table absolute core location in which the entry point is placed is indicated in the ICLT column.

When rebuilding the Executive with the SAVE ICL TABLE option, word counts and sector addresses of any interrupt core loads are retained from the old ICLT. Their interrupt assignments are indicated in the LLBB column, while the WC/EP and SA columns will contain their word counts and disk addresses. The corresponding ICLT absolute core location is found in the ICLT column.
Cold Start

Cold Start is an MPX system program. Its main function is to start your system running. There are three ways you can use the Cold Start Program: to give control to BOM, to give control to the Batch-Processing Monitor, or to give control to one of your coreloads.

Cold Starting a Batch-Processing System

Once the system-generation procedure has been completed and your batch-processing MPX system has been built, cold start can be used to set up and start that system. First Cold Start loads BOM from disk to main storage. It then establishes the logical numbers of 1810s, 2311s, and communications adapters specified on the cold start name card (described later in this chapter). Cold Start then branches to an entry point in BOM for the completion of the initialization process. BOM storage protects certain areas of itself, brings specified disks on line, starts Timer C, and prints the following Batch-Processing Monitor initialization messages:

IBM 1800 DACS MPX/BOM 00,000 00 JAN 00
SEN SW 0 ON ABSOLUTE LOADER
SEN SW 1 ON LOAD BP MONITOR
SEN SW 2 ON SET CLOCK VIA DATA SWS
SEN SW 3 ON SET DATE VIA DATA SWS

Once the above messages have been printed, the cold start operation is complete and the Batch-Processing Monitor is in control.

Cold Starting a Real-Time System

Once the system-generation procedure has been completed and your real-time system has been built, Cold Start is used to set up and start that system. Cold Start begins by loading the Executive from disk into main storage. It then establishes the logical numbers for 1810s, 2311s, and communications adapters specified on the cold start name card (described later in this chapter). Certain areas of the Executive are then storage protected, specified disks are brought on line, and reload information (if specified) is set up in the Executive. Cold Start then requests time, date and data switch settings by the following messages:

ENTER TIME THROUGH DATA SWITCHES
TIME ENTERED WAS 00.533 HOURS

ENTER DATE THROUGH DATA SWITCHES
DATE ENTERED WAS FEB 04 1970
SET VALUE IN DATA SWITCHES IF REQUIRED BY COLD START CORE LOAD

The information you set is placed in the Executive by Cold Start. Timer C is started. The cold start coreload is initiated and control is passed to the Executive. If no cold start coreload is named, background processing begins.
USING A COLD START CORELOAD

Cold starting by giving control to one of your coreloads can be performed only under a real-time system. The functions of Cold Start then are identical to those for other cold starts of a real-time system except that before calling EXIT, Cold Start queues the coreload (called the Cold Start coreload). The first operation of the real-time system will be to execute that coreload.

Cold Start Card Formats

COLD START LOADER CARDS

The seven cold start loader cards are provided by IBM along with the system-generation PID disk(s). The Loader cards constitute a program which will load the Cold Start program from disk (the last two cylinders of a physical 1810 drive or the last four tracks of a mapped 1810 drive) into main storage and branch to its entry point. (See Figure 9 for the formats of the seven cold start loader cards.)

COLD START NAME CARD

You must punch the cold start name card. Its format is shown below. It is from the cold start name card that Cold Start gets the information necessary to start a system.

Cold Start Name Card Format

<table>
<thead>
<tr>
<th>Card Column</th>
<th>Description of Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-6</td>
<td>*CLDST</td>
</tr>
<tr>
<td>7</td>
<td>Blank</td>
</tr>
<tr>
<td>8-12</td>
<td>One of the following (left-justified):</td>
</tr>
<tr>
<td></td>
<td>• BOM, for cold starting a batch-processing system</td>
</tr>
<tr>
<td></td>
<td>• BPMON, for cold starting a real-time system doing background processing</td>
</tr>
<tr>
<td></td>
<td>• NAME, the name (up to five characters long) of the cold start coreload</td>
</tr>
<tr>
<td>13,15,16,17,19,21,23,25,27,29,31,33</td>
<td>Blank</td>
</tr>
<tr>
<td>18,20,22,24,26,28,30,32</td>
<td>Numbers that specify the assignment of logical drive numbers to 1810 drives. The physical drive numbers, if any, entered in these columns become logical 1810 drives 0-7, respectively. If the drive is a mapped 1810, then enter the identifier, x, in the name of the mapped drive (BULKx). Only the numbers 0-7 may be used. Note that this assignment specifies the drive to be on the system; you cannot use the drive if you do not specify it here.</td>
</tr>
</tbody>
</table>
FIGURE 9. Cold Start Loader Cards (cont.)
This page intentionally left blank.
Card Column | Description of Contents
--- | ---
34-38 | The name of the coreload to be called in the event of a system reload (left-justified). The reload coreload must be built to execute in VCORE. If these columns are blank, the system programs an interrupt to the level and bit named in the LDLEV and LDBIT equates.
39 | Blank
40-43 | Level and partition numbers of the cold start coreload (LLAA), in decimal. Blanks are assumed to be zeros; zeros denote the basic level and VCORE.
44-47 | A four-digit hexadecimal number used, bit-by-bit, to indicate the mask status of interrupt levels 0-13 for the cold start coreload (the first hexadecimal digit corresponds to levels 0-3, the second to levels 4-7, and so on; a 1 bit indicates a level is masked). If this entry is blank, all levels are unmasked. For example, an entry of /A014 would mean levels 0, 2, 11 and 13 were to be masked, and all others unmasked. The disk interrupt level must be unmasked.
48-51 | Same as columns 44-47 except that this field indicates the mask status for interrupt levels 14-23 for the cold start coreload. These columns are meaningful only if the NULEV equate indicates 14 or more levels.
52 | Blank
53-56 | Same as columns 44-47 except that this field indicates the mask status of interrupt levels 0-13 for the reload coreload.
57-60 | Same as columns 44-47 except that this field indicates the mask status of interrupt levels 14-23 for the reload coreload. These columns are meaningful only if the NULEV equate indicates 14 or more levels.
61 | Blank
62, 63, 64, 65, 66, 67, 68, 69 | Numbers that specify the assignment of logical drive numbers to your 2311 drives. The physical drive numbers, if any, entered in these columns become logical 2311 drives 0-7, respectively. The physical drive number entered must have been previously defined (by BOM equate cards) as part of the system.
70 | Blank
71, 72, 73, 74, 75, 76, 77, 78 | Numbers that specify the assignment of logical line numbers for your communications adapters. The physical line numbers entered in these columns become logical lines 0-7, respectively. Each referenced physical line number must have been previously defined (by BOM equate cards) as part of the system. If physical lines were defined in BOM and they are to be used, these columns must be set to define the line numbers.
79 | Blank
80 | If any 2311 disk drives are used, this column must contain the hexadecimal address of the 2841 control unit, in hexadecimal. Valid entries are 0-F.

**Operating Procedures**

To cold start the MPX system, perform the following steps:

1. Turn ON the WRITE STG PRT BITS switch, turn OFF the CHECK STOP switch and set the console mode switch to RUN.
2. Clear main storage to zeros.

3. If the Cold Start Program resides on a physical 1810 drive, set data switch 0 OFF and
   enter the physical 1810 drive number in data switches 1-15 as follows:

   All data switches OFF means drive 0
   Data switch 15 ON means drive 1
   Data switch 14 ON means drive 2

   If the Cold Start Program resides on a mapped 1810 (on a 2311 drive),
   a. Set data switch 0 ON.
   b. Enter the 1810 drive identifier in data switches 4-7 (the identifier is the 1810
      number x in BULKx, the name used when the mapped 1810 drive was created).
   c. Enter the 2841 control unit address in data switches 8-11.
   d. Enter the physical drive number of the 2311 in data switches 13-15.

   For example, data switches set to

   0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
   0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0

   (where x means ON) specifies that:

   Cold Start resides on a 2311 disk drive
   The mapped 1810 drive number is 2
   The 2841 Control Unit address is A
   The physical drive number of the 2311 is 0

4. Place the seven cold start loader cards followed by the cold start name card and a blank
   card (or the following job) in the 1442 card read punch hopper and ready the reader.
   The 2311 drives must be ready before pressing program load.

5. Press console RESET and PROGRAM LOAD
   a. If this is a batch-processing system cold start, BOM is read into main storage
      from disk and the operation is complete when the BOM initialization messages
      are printed on the 1053.
   b. If this is a real-time system cold start, the Executive is read into main storage
      from disk and the following message is printed on the 1053:

      ENTER TIME THROUGH DATA SWITCHES

6. Enter the time in hours and minutes (HHMM) in the data switches and press START.
   After Cold Start reads the time from the data switches, it prints the time in hours
   and thousandths of an hour.

   TIME ENTERED WAS 00.533 HOURS

   Then the following is printed:

   ENTER DATE THROUGH DATA SWITCHES

7. Enter the date in month, day and year (MDDY) in the data switches as follows:

   Month is a hexadecimal value 1-C entered in data switches 0-3
   Day is a hexadecimal value 1-1F entered in data switches 7-11
   Year is a hexadecimal value 0-E entered in data switches 12-15
   which represents 1970-1984
When the date has been entered, press START and the following is printed:

**DATE ENTERED WAS** FEB 04 1970

8. Cold Start then prints the following message:

**SET VALUE IN DATA SWITCHES IF REQUIRED BY COLD START CORE LOAD**

If the cold start coreload requires that the data switches be set to a certain configuration, set the switches to that value.

9. Press START

This starts a real-time system as follows:

a. If this is a cold start to BPMON, background processing begins,

b. If this is a cold start to a coreload, the coreload is queued for execution.

**Cold Start Messages**

Each cold start message, whether error, informational, or instructive, has an associated indicator which is loaded into the accumulator if the 1053 specified in the BOM equate SLORG is not ready. The system then comes to a wait. Thus you can interpret certain problems that arise, correct them, and complete the cold start even though the 1053 is inoperative.

To ignore a message and continue, turn ON sense switch 0 and press console START.

To retry the cold start operation, ready the 1053, leave sense switch 0 OFF and press console START. If the reattemp is unsuccessful, the system will again wait with the indicator value in the accumulator.

When the indicator in the accumulator reflects a message requesting that you enter data in the data switches, you must enter that data at that wait; the normal data entry wait is bypassed.

The following are the informational and instructive messages printed by Cold Start. For the error messages see the IBM 1800 Multiprogramming Executive Operating System Error Messages and Recovery Procedures, Order Number GC26-3727. The hexadecimal value beside the cold start message is the indicator that will be placed in the accumulator.

**9971: WARNING PUSH START TO CONTINUE**

This message is simply a warning of a possible error. If you decide this message indicates that you need a new cold start card, then you must cold start again. To bypass the warning, press START.

**998E: DATE ENTERED WAS MMM DD 19YY**

You have entered the date shown.

**998F: ENTER DATE THROUGH DATA SWITCHES**

Enter the date in month, day, and year in the data switches and press START.

**9991: NO RELOAD CORELOAD ON NAME CARD. CALL EXIT EXECUTED AT RELOAD.**

There is no reload coreload name specified on the name card. The word count and sector address of the reload coreload in the reload information table are set to zero, indicating that a CALL EXIT is to be done during a reload.
Enter the time in hours and minutes (HH, MM) in the data switches and press START.

You have entered the time shown.

Set up the data switches as required by the cold start coreload.
BOM/Executive Reload

The reloading of a batch-processing system or a real-time system is essentially the restarting of that system after a severe error occurs. A reload differs from a cold start or a loading of BOM from cards in that BOM or the Executive is already in main storage and many of the system indicators have already been set. The reload process merely reinitializes indicators in the system and starts the system running.

The reload function is automatically performed by both the batch-processing system and the real-time system when a severe error condition is encountered while executing. An example of such an error condition is a storage protect violation. See IBM 1800 Multiprogramming Executive Operating System: Error Messages and Recovery Procedures, Order Number GC26-3727, for additional errors that would cause an automatic reload.

If you wish to stop the current execution of your system and restart it, you can do so without cold starting or loading BOM by performing a manual reload. To perform a manual reload, turn off sense switches 0 and 1, and press console STOP, RESET and START.

BOM Reload

When an automatic or manual reload of a batch-processing system is performed, the following BOM initialization messages are printed:

IBM 1800 DACS MPX/BOM XX.XXX DD MMM YY  
SEN SW 0 ON ABSOLUTE LOADER  
SEN SW 1 ON LOAD BP MONITOR  
SEN SW 2 ON SET CLOCK VIA DATA SWS  
SEN SW 3 ON SET DATE VIA DATA SWS

where XX.XXX is the time in hours and thousandths of an hour and DD MMM YY is the current date (day, month and year, respectively).

By setting the appropriate sense switch and pressing console START, you can select one of the following batch-processing options:

ABSOLUTE LOADER

Setting sense switch 0 ON and pressing START causes absolute programs in card form to be loaded by the BOM Absolute Loader from the 1442 card read punch into main storage. BOM card utilities and the System Loader are loaded in this way. BOM loads the absolute program and then branches to the entry point of the absolute program to begin its execution.

BATCH-PROCESSING MONITOR

Setting sense switch 1 ON and pressing START causes BOM to read the batch-processing Monitor Supervisor from disk into main storage. The Supervisor then begins searching the input stream for a // JOB statement.
SET CLOCK

To set the clock, set sense switch 2 ON and set data switches 0 through 15 to the time, in hours and minutes, as follows:

Hours:  
First digit: Switches 0-3  
Second digit: Switches 4-7

Minutes:  
First digit: Switches 8-11  
Second digit: Switches 12-15

For example, to set the time to 6:17 pm, enter 1817 in the data switches as follows:

0001 1000 0001 0111
1 8 1 7

Press START. This causes BOM to reset the clock to the value in the data switches and to print the BOM initialization messages again with the new time:

IBM 1800 DACS MPX/BOM 18.283 00 JAN 00
SEN SW 0 ON ABSOLUTE LOADER
SEN SW 1 ON LOAD BP MONITOR
SEN SW 2 ON SET CLOCK VIA DATA SWS
SEN SW 3 ON SET DATE VIA DATA SWS

SET DATE

To set the date, set sense switch 3 ON and set the data switches as follows:

Month  
(Hexadecimal 1-C): Switches 0-3

Day  
(Hexadecimal 1-1F): Switches 7-11

Year  
(Hexadecimal 0-E): Switches 12-15

Note that year values of 0-E correspond to 1970 - 1984, respectively.

For example, to set a date of December 22, 1970, enter C160 in the data switches as follows:

1100 0001 0110 0000
C 1 6 0

Press START. This causes BOM to reset the date to the value in the data switches and to print the BOM initialization messages reflecting the new date:

IBM 1800 DACS MPX/BOM 18.392 22 DEC 70
SEN SW 0 ON ABSOLUTE LOADER
SEN SW 1 ON LOAD BP MONITOR
SEN SW 2 ON SET CLOCK VIA DATA SWS
SEN SW 3 ON SET DATE VIA DATA SWS
Executive Reload

When an automatic or manual reload of a real-time system is done (Executive reload), the following is performed:

1. Reread from disk the user-included subroutines in the Executive if this option was elected at BOM assembly (BOM equate SYEXR set to 1).

2. Reread from disk the SPAR coreload(s) currently in main storage if this option was elected at BOM assembly (BOM equate RLSPR set to 1).

3. Unmask the system to the mask condition specified in the cold start name card. If a reload coreload was specified at cold start, load this coreload from disk to VCORE and branch to it (BOM equate NOREL must be set to 1). If no reload was specified, program an interrupt to the level and bit named in the LDLEV and LDBIT equates.
Background-Processing Operations

Background-processing operations consist of assemblies, compilations, Disk Management Program functions, Builder functions, and background program executions. They run under the control of the Batch-Processing Monitor Supervisor (SUP). SUP reads and analyzes all Supervisor control statements, such as // JOB, // ASM, and // FOR, initializes the Disk Communications area (DCOM) when a // JOB statement is encountered, loads requested programs, and passes control to them.


Loading the Supervisor

The Supervisor can run in a batch-processing system (under BOM) or in a real-time system (under the Executive).

To load and pass control to the Supervisor under a batch-processing system, BOM must first be initialized. BOM can be initialized in any of three ways:

- Loading the BOM object card deck to main storage with the BOM high core loader (see "System Generation").
- Cold Starting to BOM (see "Cold Start").
- Reloading BOM (see "BOM/Executive Reload").

When BOM is initialized, the following messages are printed:

IBM 1800 DACS MPX/BOM XX,XXX YY MTH ZZ
SEN SW 0 ON ABSOLUTE LOADER
SEN SW 1 ON LOAD BP MONITOR
SEN SW 2 ON SET CLOCK VIA DATA SWS
SEN SW 3 ON SET DATE VIA DATA SWS

Setting sense switch 1 ON and pressing console START causes BOM to load and pass control to the Supervisor.

Note that setting sense switch 7 ON and pressing CONSOLE INTERRUPT at any time during execution of any program will cause that program to be aborted, the Supervisor to be loaded, and control to be passed to the Supervisor.

To load and pass control to the Supervisor under a real-time system, do a cold start to BPMON (see Cold Start). Background processing will then be carried out until a // END statement is encountered. Such a statement signals the Supervisor that there is no more background processing. To start background processing again after a // END statement has been encountered, set sense switch 7 ON and press CONSOLE INTERRUPT. This will cause the Supervisor to be loaded and given control.

Input to the Supervisor

Once SUP has control, it begins searching the input stream for a // JOB statement. A // JOB statement causes SUP to terminate any previous background-processing job and to begin a new one.
Other control statements must immediately follow a //JOB statement in the input stream to direct the Supervisor to load the proper program (Macro Assembler, FORTRAN Compiler, etc.). For a detailed description of the MPX control statements, see IBM 1800 Multiprogramming Executive Operating System Programmer's Guide, Order Number GC26-3720. For a summarized description of the control statements, refer to MPX Control Statements, Order Number GX26-1594.

The input stream to the Supervisor may be from the 1442 card read punch or the 1816 keyboard. The card reader is the assumed input source unless otherwise specified.

To specify the 1816 keyboard as the input device, the //SET control statement may be used (see IBM 1800 Multiprogramming Executive Operating System Programmer's Guide, Order Number GC26-3720, or MPX Control Statements, Order Number GX26-1594). The other way to change the input stream device is to press the 1816 keyboard request key with sense switch 7 ON. This will cause execution of the current background-processing operation to be terminated.

Background processing initiated by keyboard request will be terminated after a time lapse of five minutes if no input from the 1816 occurs. Note that this time lapse has the same effect as a //END statement.

When the keyboard is being used as the input stream device, you can go back to using the 1442 in two ways:

- Press CONSOLE INTERRUPT with sense switch 7 ON. This will cause the current background-processing job to be aborted and background processing to continue through input from the card reader.
- Using the //SET statement to transfer the input stream device assignment without terminating the current job.

KEYBOARD INPUT

Input through the 1816 keyboard is accepted by the Disk Management Program as well as the Batch-Processing Monitor Supervisor. That input can consist of any control statements acceptable to these programs with the exception of the *CCEND statement required to end a store operation. That statement must be entered by the card reader. Non-binary input following *MON and *SRFLE control statements may also be entered through the keyboard.

Once the input stream device has been established to be the 1816 keyboard, the green PROCEED light on that keyboard comes on. The Supervisor is then waiting for your input. Your input control statements should be typed in print locations corresponding to card columns. When you have typed the control statement, press the EOF key to send the statement to the Supervisor.

CORRECTION OF ERRORS

If you make a mistake before pressing the EOF key, you may use the ER CHR key to backspace and re-enter a character. Alternatively, you may press the ER FLD key and re-key the entire statement. When the Supervisor or Disk Management Program encounters an error in the control statement, an error message will be typed on the system printer. Other system messages such as DMP FUNCTION COMPLETE will also be printed on the system printer. Note that the system printer you defined at BOM assembly (by BOM equate cards) is not necessarily the 1816 currently being used as your input device. The //SET statement should be used if you want to alter printer assignments.
CHECKING INPUT STATEMENTS

If you want to check an input statement, type two question marks (??) at the end of that statement. This must be done before the EOF key is pressed. The string just entered is typed in red so you can check it. If the statement is acceptable, press EOF. If the statement is unacceptable, cancel it by pressing the NOT sign (→) key.

// FOR NAME ?? (Typed input)
// FOR NAME (Typed output in red)

REQUESTING STATEMENT FORMATS

If you are unfamiliar with the format of the control statement you wish to use, enter a question mark (?) followed by a blank and the name of the control statement and press the EOF key. This will cause the standard format of the specified control statement to be typed, with question marks replacing asterisks and slashes. You can then type your input directly below the line displayed.

? DUMP (Typed input)
?DUMP Z XXN YYM NAMEP NSEC (Typed output in red)
*DUMP UA PR RW15D (Typed input)

DMP FUNCTION COMPLETED

You must know the correct name for the control statement. The message INCORRECT OR INSUFFICIENT INPUT. TRY AGAIN is typed if the control-statement name is not correct.

MACRO ASSEMBLER LISTING

This listing shows typical output from the Macro Assembler. The circled numerals on the listing are keyed to the notes that follow the listing.

// JOB 09 JUL 70 10.974 HRS
// ASM M 09 JUL 70 10.975 HRS
*LIST
*XREF
*ONE WORD INTEGERS
*COMMON REAL,INT,ARRAY(2,2,2),IRAY(5,10)
*COMMON/INSKEL/ AREAD,R,1B
0020 33 M MULTI DC 3
  2ND ATTEMPT TO DEFINE MULTI
0000 03 C400FFE 1
  LD L REAL
0002 03 C400FFFD 2
  LD L INT
0004 03 C4000000 3 R
  LD L REAL+INT
0006 02 C400FFFCC 4
  LD L B
0008 02 C400FFF8 5
  C4 EQU 
000A 02 C4000000 7 R
  LD L IB+8+4
000C 02 C400FFFF 8
  LD L IB+4
FFFE 9 AA EQU REAL
000E 03 C400FFE 10
  LD L AA
FFFD 11 S SET INT
0010 03 C400FFFD 12
  LD L S
0012 0 C000 13 R
  LD L S
0013 0 62FO 16
  LOOP LDX 2 -16
0014 03 CE00FFFC 17
  LOOP LD L ARRAY+2 LOAD EXTENDED PREC WORD
0016 0 7202 18
  MDX 2
0017 0 70FC 19
  MDX LOOP
0018 0 61CE 20
  LDX 2 -16
0019 03 C500FFFC 21
  LOOP LD L LOP LOAD INTEGER ELEMENT
0018 0 7101 22
  MDX 1
001C 0 70FC 23
  MDX LOOP
001D 0 C007 24
  LD =REAL LOAD ADDR OF COMMON ELEMENT
001E 0 C007 25
  LD =12 CONSTANT OF 12
001F 0 0001 26
  MULTI DC 1
  FIRST DEFINITION OF MULTI
0020 0 0003 27
  MULTI DC 3
  2ND ATTEMPT TO DEFINE MULTI
0021 01 C4000001F 28
  LD L MULTI NOTICE NO M FLAG
0023 00 C4000000 29
  UD L UNDEF REF TO UNDEFINED LABEL
0025 30 0
  GO EQU 
  DEFINE LITERAL TABLE
0025 31
  LOOP LOP LOAD EXTENDED PREC WORD
0025 32
  LOOP LDP LOAD INTEGER ELEMENT
0025 33
  LOOP LOOP LOAD EXTENDED PREC WORD
0025 34
  LOOP LOOP LOAD INTEGER ELEMENT
0025 35
  LOOP LOOP LOAD INTEGER ELEMENT
0025 36
  LOOP LOOP LOAD INTEGER ELEMENT
0025 37
  LOOP LOOP LOAD INTEGER ELEMENT
0025 38
  LOOP LOOP LOAD INTEGER ELEMENT
0025 39
  LOOP LOOP LOAD INTEGER ELEMENT
0025 40
  LOOP LOOP LOAD INTEGER ELEMENT
0025 41
  LOOP LOOP LOAD INTEGER ELEMENT
0025 42
  LOOP LOOP LOAD INTEGER ELEMENT
0025 43
  END GO
005 ERROR(S) AND 000 WARNING(S) IN ABOVE ASSEMBLY.

SYMBOL VALUE REL DEFN REFERENCES-
AA FFFE C-COMMON 10R
ARRAY FFFE I-COMMON
ARRAY FFFA C-COMMON 17R
B FFFC I-COMMON 4R 6R 7R
c 0004 0 5
0004 0004 5
0004 0004 5
0025 0025 358 428 43R
0025 0025 38 7R 8R
INT FFFD C-COMMON 2R 3R 11R
IRAY FFEH C-COMMON 24R
LOOP 0014 117 19R
LDP 0019 1 24 26M
MULTI MULT DFN/SET 32 35R
REAL FFFE C-COMMON 1R 3R 9R 40R
S FFFD C-COMMON 12R 13R
UNDEF **UNDEFINED-SYMBOL** 37R
Notes:

1. All statements that are found to be in error by the Macro Assembler during pass one are printed. These statements may or may not be flagged during pass two.

2. At the end of pass one, the A05 message is printed on the list and system printers if any errors were detected during that pass.

3. A 3 is printed in column 7 if an instruction references a symbol defined in coreload COMMON area.

4. An R flag is printed in column 24 if an illegal reference is made to a COMMON element.

5. A 2 is printed in column 7 if an instruction references a symbol defined in INSKEI COMMON.

6. The SET statement can be used to assign a COMMON element a new name.

7. A literal can be used to obtain the address of an element defined in a COMMON area.

8. This statement shows normal constant definition using a literal.

9. This statement is erroneous but is flagged only on pass one of the assembly.

10. The symbol MULTI is assigned the value of the first definition (line 32). Any reference to a multiply-defined symbol is not flagged with an M.

11. This statement shows normal reference to an undefined symbol. Also see the cross-reference symbol table.

12. The LORG statement is used to instruct the Macro Assembler to define the literals at this point.

13. This DC is generated to define the address of REAL.

14. This DC is generated to define the constant 12.

15. The number printed in the error message signoff includes errors diagnosed during both pass one and pass two.

16. Columns 18–23 show the line number of each statement that the Macro Assembler processes. Even statements such as EJCT, LIST and HDNG which are not printed are assigned a line number.

17. A + is always printed in this column for a Macro Assembler-generated statement.

18. The cross-reference heading is printed at the top of each page of the XREF table. The column under SYMBOL shows each symbol defined in an assembly; under VALUE is shown the value assigned to the symbol during the assembly; under REL is shown a 0 or 1 to indicate whether the symbol is absolute or relocatable; under DEFN is shown the number of the line where the symbol is defined in the assembly (or see 19 and 20); under REFERENCES is shown the number of the line where each symbol is referenced. The only references that will not be found in this table are from AIF and AGO statements assembled in one-pass mode. (See also 23.)

19. A symbol defined in the coreload COMMON area is assigned the code C-COMMON in the column under the heading DEFN.

20. A symbol defined in INSKEI COMMON is assigned the code I-COMMON in the column under the heading DEFN.

21. A symbol which is either multiply-defined or defined more than once by the SET statement is assigned the code MULT DFN/SET in the column under the heading VALUE REL.
22. A symbol that is referenced but not defined in an assembly is assigned the code **UNDEFINED-SYMBOL**.

23. With each reference line number printed under the heading column REFERENCES-, one of the three letters B, M, and R is printed to indicate that the referenced instruction is of the type branch, modify, or reference. For example, a BSC is a branch, an MDX or STO is a modify, and an LD or A is a reference. A short MDX reference will show an M even though it is a branch.
DMON-System Maintenance Program

This program does real-time maintenance on the MPX system. It operates under DMP and is called by a *MON statement.

There are instances when IBM requires that you do a DMON operation (for example, to install a PTF, change the Version or Modification number of the system, or load a new version of a system program), and there are instances when you may choose to do one yourself (for example, to patch a particular disk word, delete the Macro Assembler and/or FORTRAN Compiler from disk, or modify a system program).

DMON can do the following:

- Patch any word in a disk sector.
- Patch system programs (patching outside current program disk boundaries is permitted).
- Install Program Temporary Fixed (PTFs) to system programs and record them in a log.
- Print the PTF log.
- Update the Version and Modification level of the system.
- Delete the FORTRAN Compiler and/or Macro Assembler from disk and pack the disk.
- Load a new version of a system program already on disk. (New subroutines are loaded by *STOREMD. *MON is used to update the Version/Modification number and the PTF log.)

In summary, DMON permits system updating, keeps a record of the status of the system, prints this status on demand, and prints out a complete picture of all patches installed.

A card deck containing corrections to update the MPX system to its latest Version and Modification level is supplied by IBM. Each time you receive one of these decks you must install the update by running the deck, even if the program affected has been deleted from your MPX system. This is necessary in order to update your system version and modification level.

The calling sequence for DMON is:

```
1-10  11-20  21-30  31-40  41-50  51-60
1234567890  1234567890  1234567890  1234567890  1234567890  1234567890
// JOB
// DMP
*MON
```

The entries in the *MON control statement fields determine the function to be done by DMON. Any number of *MON control statements can be used in a DMON operation.
The following is the general format of the *MON control statement:

```
MON VNMM XXXX NNNN PPPP DSSS CCCC 1800-8 XXXX-BPPP-XXXX-SXXX-SXXX-SXXX
```

where:

- **VNMM** (columns 6-9) indicates that this PTF is applicable to MPX systems of Version V and Modification levels from MM through MM+N. The numbers must be specified in hexadecimal format. If column 7 contains a U, the system Version and Modification level numbers are updated to the values specified by V and MM.

- **XXX** (columns 11-14) is the program name (left-justified) as it appears in LET. (If this field is blank, an absolute patch (see columns 28-31 and 33-36) is being done.)

- **NNNN** (columns 16-19) specifies the number of *MON statements including this statement, in decimal. This field is used only on the first *MON statement of each DMON function. Leading zeros must be entered.

- **Z** (column 21) specifies the type of patch records that will follow the *MON statement. It must be either H, which indicates hexadecimal patches, or B, which indicates binary patches.

- **PPPP** (columns 23-26) specifies the number of patch records following this MON statement, in decimal. Leading zeros must be entered.

- **DSSS** (columns 28-31) specifies the drive number and sector address, in hexadecimal. This field is used only if the LET entry field (columns 11-14) is blank. The DSSS field is used for patching a relative word or a program on the specified sector.

- **CCCC** (columns 33-36) specifies the address that would be occupied by the first word of the sector specified in columns 28-31 if it were loaded into main storage. This field is used only if columns 28-31 are used. If this field is blank and columns 28-31 are used, a relative word in the sector is being patched.

The remaining fields of the *MON statement are used to modify, maintain, and print the PTF log.

- **1800-8** (columns 38-43) is the PTF sign-on indicator to indicate that a PTF is to be installed. This entry is also used when a printout of the PTF log is desired.

- **KKKK** (columns 44-47) specifies a number unique to the PTF being installed. If this field is 0000, a PTF is being removed from the PTF log (see columns 52-71).

- **-RRR** (columns 48-51) is reserved for IBM program support code. The first column in this field must contain a minus sign.
SKKKK (columns 52-71) these fields are used to provide control information for PTF installation and recording. They serve three purposes:

1. They indicate whether prior PTFs are prerequisite to the PTF currently being installed or removed. A prerequisite PTF is indicated when the S part of a field is blank. The KKKK part is the PTF number, in decimal.

2. They indicate whether prior PTFs are made obsolete by the PTF currently being installed. A PTF made obsolete is indicated when the S part of the field contains a minus sign. The KKKK part contains the PTF number, in decimal.

3. If the S part of a field contains a minus sign and columns 44-47 contain zeros, the PTF number specified in the field will be removed from the log.

Any errors in the *MON control statement cause the system to abort the DMON function and return to DMP. However, an error in a patch record terminates the processing of that record only.

Each time a system is regenerated from the PID disks, all patches must be run to bring the system up to operational level. You should keep all program updates and patches in one deck so that they can be loaded by DMON.

**Patching Any Word in a Disk Sector**

If a patch function is being done, you must use an *MON statement followed by patch records in hexadecimal format or patch cards in binary format. The control statement format is as follows:

```
1-10 11-20 21-30 31-40 41-50 51-60
1234567890 1234567890 1234567890 1234567890 1234567890 1234567890
1121314151617181910111213141516171819101112131415161718191011121314151617181910
MD W.
```

Where:

**VNMM**
(columns 6-9) indicates that this PTF is applicable to MPX systems of Version V and Modification levels from MM through MM+N. The numbers must be specified in hexadecimal.

**NNNN**
(columns 16-19) indicates the number of *MON statements, including this statement, in decimal. This field is used only on the first *MON statement of each DMON function. Leading zeros must be entered.

**X**
(column 21) indicates the type of patch cards that follow. This column can contain H for hexadecimal or B for binary.

**PPPP**
(columns 23-26) indicates the number of patch records following this *MON statement, in decimal. Leading zeros must be entered.

**DSSS**
(columns 28-31) indicates the drive number and sector address to which the patch will be made.
(columns 33-36) indicates the address that would be occupied by the first word of the sector specified in the DSSS field (columns 28-31) if it were loaded into main storage. This field can only be used if columns 28-31 are used. If this field is blank and columns 28-31 are used, a relative word in the sector is patched.

HEXADECIMAL PATCH RECORD FORMAT

The format of a hexadecimal patch record is:

<table>
<thead>
<tr>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAAA</td>
<td>XXXX</td>
<td>XXXX</td>
<td>XXXX</td>
<td>XXXX</td>
<td>XXXX</td>
</tr>
</tbody>
</table>

AAAA specifies, in hexadecimal, the main-storage address of the first word to be patched. If a relative patch is specified (columns 28-31 contain a sector address and columns 33-36 are blank on the *MON control statement), the relative word number is specified. For example, 001F in this field would cause a patch of relative word 31 in the sector specified in columns 28-31 of the *MON control statement.

XXXX specifies one hexadecimal word of patch data. As many as 15 consecutive words can be specified on one patch record. Each word must be followed by a blank column.

BINARY PATCH CARD FORMAT

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Main-storage address of the first word to be patched, or the number of the relative word on the sector.</td>
</tr>
<tr>
<td>3</td>
<td>First 8 bits — the type of patch card (1, A, or F). A is the type of a normal patch card. The 1, A, or F appears right-justified in the 8-bit field. Last 8 bits — number of patches on the card.</td>
</tr>
<tr>
<td>10-54</td>
<td>Data words 1 through 45.</td>
</tr>
</tbody>
</table>

See Appendix C for a description of card types and an illustration of a binary card.

When an absolute patch is made, the patch is printed on the list printer. If data switch 15 is OFF, the system will execute a WAIT after printing each patch. If the patch is correct, push START with data switch 0 OFF. If data switch 15 is ON, the patch will be made and the system will not execute a WAIT.
Patching System Programs

DMON may be used to patch any system program or program phase. It is used to install Program Temporary Fixes (PTFs); a PTF is in the form of a patch or a new program.

A new *MON statement is required for each program or phase patched. The program sets no limits to the number of patch data records that can be used following the *MON statement.

The control statement format is:

<table>
<thead>
<tr>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td>123456789012345678901234567890123456789012345678901234567890</td>
<td>*MON VNMM XXXX NNNN Z PPPP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

VNMM (columns 6-9) indicates that this PTF is applicable to MPX systems of Version V and Modification levels from MM through MM+N. The numbers must be specified in hexadecimal.

XXXXX (columns 11-14) specifies the program name (left-justified) as it appears in LET.

NNNN (columns 16-19) indicates the number of *MON statements, including this statement, in decimal. This field is used only on the first *MON statement of each DMON operation. Leading zeros must be entered.

Z (column 21) indicates the type of patch cards that follow. This column can contain H for hexadecimal or B for binary.

PPPP (columns 23-26) indicates the number of patch records following this *MON statement, in decimal. Leading zeros must be entered.

Figure 10 illustrates a typical system program patch operation. Sector patches and PTFs are installed in the same manner.
Installing Program Temporary Fixes (PTFs) to System Programs and Recording Them in a Log

Installing PTFs to system programs is handled in the same manner as described under "Patching System Programs." However, in order to record the installation of a PTF in the PTF log the following format is required for the *MON control statement:

```
1-10  i-20  21-30  31-40  41-50  51-60

61-70  71-130
```

The VNMM, XXXX, NNNN, Z, and PPPP fields are the same as described under "Patching System Programs."

1800-8 (columns 38-43) is the PTF sign-on indicator, used to indicate that a PTF is being installed.

KKKK (columns 44-47) is the number unique to the PTF being installed. If this field is 0000, it indicates that a PTF is being removed from the PTF log (see SKKKK field).

-RRR (columns 48-51) is reserved for IBM program support code. The first column in this field must contain a minus sign.

SKKKK (columns 52-71) is a sequence of fields used to provide control information for PTF installation and recording. They serve three purposes:

1. They indicate whether prior PTFs are prerequisite to the PTF currently being installed or removed. A prerequisite PTF is indicated when the S part of the field is blank. The KKKK part contains the PTF number in decimal.

2. They indicate whether prior PTFs are made obsolete by the PTF currently being installed. An obsolete PTF is indicated when the S part of the field contains a minus sign. The KKKK part contains the PTF number, in decimal.

3. If the S part of a field contains a minus sign and columns 44-47 contain zeros, the PTF specified in the field will be removed from the log.

Printing the PTF Log

You can obtain a copy of the current PTF log by using an *MON statement containing the following entries:

```
1-10 11-20 21-30 31-40 41-50 51-60
1800-8
```

72 1800 MPX Operating Procedures
This is an example of the PTF log:

```
// JOB 02 FEB 70 10.316 HRS
// DMP 02 FEB 70 10.318 HRS
*MON
VER/MOD. 2000.
PTF NO. 0001.
PTF NO. 0002.
PTF NO. 0003. obsolete
PTF NO. 0004.
PTF NO. 0005.
PTF NO. 0006.
PTF NO. 0007.
PTF NO. 0008. obsolete
PTF NO. 0009.
PTF NO. 0010.
PTF NO. 0011.
PTF NO. 0012.
DMP FUNCTION COMPLETED
```

### Updating the Version and Modification Level of the System

DMON can be used to update the system version number and modification level number in the Disk Communications Area (DCOM). DMON updates the PTF log by recording the number associated with the PTF in the *MON control statement. The *MON control statement for this operation is as follows:

```
*MON NUMM

where:

V (column 6) is the Version number. It can be only one greater than the current Version number. If the Version number is updated, the Modification level must be zero.

U (column 7) is a Version/Modification level update indicator.

MM (columns 8-9) is the Modification level number. It can be only one greater than the current Modification level number.

BBBB (columns 44-47) is one less than the first PTF number assigned to the new Modification level.

Note: A PTF number is ignored if it is specified as a PTF to be deleted, obsoleted, or specified as a prerequisite and its value is less than the BBBB field described above.

### Deleting the FORTRAN Compiler and/or Macro Assembler from Disk and Packing the Disk

DMON may be used to delete the Macro Assembler and/or the FORTRAN Compiler from the system. Disk areas are repacked following the deletion.
The *MON statements for deleting the FORTRAN Compiler and the Macro Assembler, respectively, are:

<table>
<thead>
<tr>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>MON</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td><strong>FORTRAN</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

When such a statement is processed, the following messages will be printed:

YOU ARE DELETING FORTRAN OR THE ASSEMBLER
IF OK, PUSH START WITH DATA SWITCH 0 OFF

Deletion of the Macro Assembler or FORTRAN Compiler is not recommended for a short-term gain of disk room. These programs may be reloaded from cards using *MON (if they are available in card form), but it will take from one to three hours to reload either of them. If the programs are not available in card form, a new system generation must be performed to restore these programs to the system.

**Loading New Versions of System Programs Already on Disk**

A single *MON control statement can be used to load more than one new object program deck, so long as the programs being replaced are system programs and the last deck is followed by an *CCEND card. If the new version of a program requires more disk sectors than its previous version, DMON will enlarge the program area to make an additional sector available. This increase is limited to one sector. Similarly, if a new version of a program requires fewer disk sectors than its previous version, DMON allows packing of the program area. Packing is also limited to one sector.

Any DMON function that involves an expansion or contraction of one sector of the affected program will cause the following message to be printed:

UPDATE REQUIRES DISK MOVE

If data switch 1 is on, the program performs the move and continues. If data switch 1 is OFF, the following message is printed and the program executes a WAIT:

IF OK, PUSH START WITH DATA SWITCH 0 OFF

Turn ON data entry switch zero and press console START if you want to abort the current operation.

The control statement format for loading a new version of a system program is:

<table>
<thead>
<tr>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>MON</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>WNNM XXXX</td>
<td>WNNNPPP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
VNMM (columns 6-9) indicates that this PTF is applicable to MPX systems of Version V and Modification levels from MM through MM+N. The numbers must be specified in hexadecimal.

XXXX (columns 11-14) is the name of the program or program phase, left-justified, as it appears in LET. If the *MON statement is being used to replace two or more programs, this is the name of the first one.

NNNN (columns 16-19) is the number of *MON statements, including the statement, in decimal. This field is used only on the first *MON statement of each DMON function. Leading zeros must be entered.

Z (column 21) must be B, indicating that binary patch cards follow the *MON statement.

PPPP (columns 23-26) is a decimal number equal to or greater than the total number of object cards following the *MON control statement.

The load of a new version of a system program is terminated by an *CCEND control card; this card must follow the final (or only) object deck being loaded.

Typical input for a system program update is illustrated in Figure 11.

![Figure 11. Input for a System Program Update](image)

**Replacing Library Subroutines**

Changes to the MPX Subroutine Library require the deletion of the old program and the storing of the new one. *MON just updates the Version and Modification level word; the actual loading of the new program is performed by a DMP *STOREMD operation.

The format of the *MON control statement is:

```
MON VNMM XXXX NNNN Z PPPP
```

Replacing Library Subroutines
The fields were described in the section "Installing Program Temporary Fixes (PTFs) to System Programs."

Typical input for Subroutine Library maintenance is illustrated in Figure 12.

![Figure 12. Deck for Updating the Subroutine Library](image)

**Loading a New Version of DMON**

DMON cannot be used to replace itself. New versions of DMON must be loaded by the System Loader, under control of BOM. The following input is required to load a new version of DMON.

<table>
<thead>
<tr>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td>123</td>
<td>456</td>
<td>7890</td>
<td>123</td>
<td>456</td>
<td>7890</td>
</tr>
</tbody>
</table>

```
1. *LDDSK

2. system loader

3. /PARTIAL_SYSTEM_UPDATE

4. new version of DMON

5. CC.END
```

Data switch 14 ON will suppress all printing (except for error messages) during the loading process. When a new version of DMON is loaded, an *MON function should be performed to update the PTF log in DCOM.

Note: The sequence shown for replacing DMON is also valid for replacing any other system program.
BOM Card Utilities

A group of utility programs that can be loaded into main storage by the BOM Absolute Loader are punched during the system-generation process.

These object programs are:

- **BDCRL** - BOM 2311 Disk Dump to Cards and Reload Program
- **BDUMP** - BOM 1810 Disk Dump to Cards Program
- **BDPAT** - BOM Disk Patch Program
- **BDPIP** - BOM 2311 Disk Pack Initialization Program
- **BDUPL** - BOM Disk Duplication Program
- **BDWAP** - BOM Disk Write Addresses Program
- **BLIST** - BOM 80-80 List and Sequence Check Program
- **BRELD** - BOM 1810 Disk Dump Reload Program

During the operation of each of these programs instructions will be printed on the system printer. You will have the ability to select various options by entering data through the console switches. The functions requested and the data entered are printed during the operation of these programs.

You can return to BOM at any time by pressing console STOP, RESET, and START with all sense switches OFF.

**BDCRL - BOM 2311 Disk Dump to Cards and Reload Program**

BDCRL can dump all or part of the contents of a 1316 disk pack on drive 0 to cards, and can reload such data from cards to disk. The program can be used to dump contents from one disk pack onto cards and to then reload these contents onto another pack. Contents to be dumped can begin at any cylinder address and end at any other cylinder address.

**DUMP OPERATING PROCEDURES**

1. Load BOM into main storage.
2. Place BDCRL into the card read punch hopper followed by blank cards.
3. Execute the BOM absolute loader (sense switch 0 ON, data switch 15 ON).
4. The following message is printed when program execution begins:

```
BOM 2311 DISK DUMP/RELOAD PGM
CYLINDER ADDRESS RANGE 0000-00C7
SET DATA SW 0 ON TO DUMP
SET DATA SW 0 OFF TO RELOAD
```
5. Set data switch 0 ON.
6. Press console START.
7. The following message is printed:

```
DATA SW = STARTING CYLINDER ADDRESS
```
8. Set the address of the first cylinder to be dumped in the data switches.
9. Press console START.
10. The following message is printed:

```
DATA SW = LAST CYLINDER ADDRESS
```
11. Set the address of the last cylinder to be dumped in the data switches.
12. Press console START.
The program will now begin the dump of the 1316 disk pack to cards. When the dump is complete, the program returns to step 4. Press console STOP, RESET, and START to end the BDCRL operation.

The volume label record (cylinder 0, track 0, record 3) contains a pointer to the Volume Table of Contents (VTOC). If cylinder 0 is not dumped, the VTOC is assumed to be at the disk address currently indicated by the existing volume label record on the disk to be reloaded. You normally should dump cylinder 0 when dumping the VTOC (any time you dump less than the entire disk).

RELOAD OPERATING PROCEDURES

1. Load BOM into main storage.

2. Place BDCRL into the card read punch hopper followed by the cards to be reloaded (obtained from a previous BDCRL disk dump to cards) and two blank cards.

3. Execute the BOM Absolute Loader (sense switch 0 ON, data switch 15 ON).

4. The following message is printed when program execution begins:

   BOM 2311 DISK DUMP/RELOAD PGM
   CYLINDER ADDRESS RANGE 0000-00C7
   SET DATA SW 0 ON TO DUMP
   SET DATA SW 0 OFF TO RELOAD

5. Set data switch 0 OFF.

6. Press console START.

7. The following message is printed:

   DATA SW = LAST CYLINDER ADDRESS

8. Press console START.

When the reload is complete, the program returns to step 7. Press console STOP, RESET, and START to end the BDCRL operation.

CARD FORMAT

The format of data on cards is Card Data (CDD) Format. Each new record begins on a new card. The first word of each new card contains zero. The formats of the other cards are as follows:

First Cylinder Break Card

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/FFFF</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Reserved</td>
</tr>
<tr>
<td>4</td>
<td>Cylinder Address</td>
</tr>
<tr>
<td>5</td>
<td>/FFFF</td>
</tr>
<tr>
<td>6-59</td>
<td>Reserved</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>
Subsequent Cylinder Break Cards

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/FFF</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Reserved</td>
</tr>
<tr>
<td>4</td>
<td>Cylinder Address</td>
</tr>
<tr>
<td>5-59</td>
<td>Reserved</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>

Cylinder Data Card

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Relative data word address to the beginning of the record</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Word count of number of data words</td>
</tr>
<tr>
<td>4-59</td>
<td>Data words</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>

End-of-All-Cylinder Cards

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reserved</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>/FFFF</td>
</tr>
<tr>
<td>4-59</td>
<td>Reserved</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>

**BDUMP - BOM 1810 Disk Dump to Cards Program**

This program dumps all or part of the contents of 1810 drive 0 onto cards. Drive 0 may be physical or mapped. The file protect status of each sector is included in the dump. The data can be reloaded to a new 1810 drive by using the BOM 1810 Disk Dump Reload Program (BRELD).

**OPERATING PROCEDURES**

1. Load BOM into main storage.
2. Place BDUMP into the card read punch hopper followed by blank cards.
3. Execute the BOM Absolute Loader (sense switch 0 ON, data switch 15 ON).
4. The following message is printed when program execution begins:

```
BOM DISK DUMP TO CARDS DRIVE 0
CYLINDER ADDRESS RANGE 0000-01FF
DATA SW = FIRST CYLINDER ADDRESS
```
5. Set the address of the first cylinder to be dumped in the data switches.
6. Press console START.
7. The following message is printed:

```
DATA SW = LAST CYLINDER ADDRESS
```
8. Set the address of the last cylinder to be dumped in the data switches (maximum of 0007 for a physical 1810 drive, maximum of /01FF for a mapped 1810 drive).

9. Press console START.

The program will now begin the dump of the 1810 drive to cards. When the dump is complete, the program returns to step 4. Press console STOP, RESET, and START to end the BDUMP operation.

CARD FORMAT

The output of the BOM Disk Dump to Cards Program is in binary format, 3/4 of a 16-bit binary word to a card column. This format allows sixty data words to be punched in each 80-column card (1-1/3 columns equal one binary word). (Appendix C shows the layout of a binary card.) The disk data output is in the following format:

Cylinder Break Card

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/FFFF</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Reserved</td>
</tr>
<tr>
<td>4</td>
<td>Sector address</td>
</tr>
<tr>
<td>5</td>
<td>File protect status (/8XXX=on)</td>
</tr>
<tr>
<td>6-59</td>
<td>Reserved</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>

Cylinder Data Card

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Relative address of data on cylinder</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Word count of number of data words</td>
</tr>
<tr>
<td>4-59</td>
<td>Data words</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>

End-of-All-Cylinder Cards

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reserved</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>/FFFF</td>
</tr>
<tr>
<td>4-59</td>
<td>Reserved</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>

BDPAT - BOM Disk Patch Program

This program allows you to alter any data words on any 1810 drive (physical or mapped) in the system using disk address cards and data cards loaded as card sets.

A card set is defined as a disk address card followed by any number of data cards. The data cards cause the specified words to be modified. Any number of card sets may be loaded by the disk patch program.
OPERATING PROCEDURES

1. Load BOM into main storage.

2. Place BDPAT into the card read punch hopper followed by the card sets and two blank cards.

3. Execute the BOM Absolute Loader (sense switch 0 ON, data switch 15 ON).

4. The following messages are printed when program execution begins:

```
BOM DISK PATCH PROGRAM
DISK ADDR ORIGIN IS AAAA
DATA ADDR ORIGIN IS BBBB
ABSA SECA RELA OLDV NEWV
CCCC DDDD EEEE FFFF GGGG
```

Note that the patch has already taken place when these messages are printed.

AAAA is the drive number and sector address of the first sector of the disk area being patched.

BBBB is the main-storage address that would be occupied by the first data word of the area being patched if it were loaded into main storage.

CCCC is the main-storage address of the word being patched.

DDDD is the address of the sector actually being patched.

EEEE is the relative address within the sector of the word being patched.

FFFF is the old value of the patched word.

GGGG is the new value of the patched word.

If an invalid data card or a blank card is read, the program prints the following message and executes a WAIT:

```
DATA CARD ERROR OR
NO MORE DATA CARDS
THEREFORE JOB HALTED
```

If the WAIT is due to a data card error, correct the card and place the entire card set, starting with the disk address card, back into the card read punch hopper. Press reader START and console START. If the WAIT is not due to a data card error, you may place more card sets in the hopper and press console START to continue, or you may turn off all sense switches and press console STOP, RESET, and START to return to BOM.

CARD FORMAT

Disk Address Card

<table>
<thead>
<tr>
<th>Columns</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>*(An asterisk signifies a disk address card)</td>
</tr>
<tr>
<td>2-5</td>
<td>Drive number and first sector address (DSSS) of disk area to be patched (hexadecimal disk addresses are listed in Appendix B)</td>
</tr>
<tr>
<td>6</td>
<td>Blank</td>
</tr>
</tbody>
</table>
The 2311 BOM Disk Pack Initialization Program in card form (BDPIP) is one of the card utilities punched during system generation.

The 2311 BOM Disk Pack Initialization Program functions according to 2311 initialization control statement specifications. The program has two functions:

- An *INITLZ control statement causes the program to initialize the 1316 disk pack. Initialization includes the following:
  - Optional checking of the existing volume serial number against the one specified in the control statement. The remaining *INITLZ functions are aborted if the volume serial numbers do not match.
  - Checking for defective tracks, with or without an optional surface analysis, with automatic assignment of alternates for those tracks found to be defective.
  - Writing a volume label record on the pack.
  - Constructing and writing the Volume Table of Contents (VTOC).
- An *GETALT control card causes the program to do the following:
  - If a volume serial number is specified, check it against the existing volume serial number and abort remaining *GETALT functions if the numbers are not identical.
  - Assign an alternate track without testing to see if the track is defective.
  - Test a track and, if it is defective, assign an alternate.

The formats of the control statements follow.

The format of *END BDPIP is:

```
Columns Value

1-10 Main-storage address of the word being patched
11-20 Not used

1-10 Main-storage address of the first data word of the area being patched
11-80 Not used

Data Card

Columns Value

1-4 Main-storage address of the word being patched (must be equal to or greater than the entry in columns 7-10 on the disk address card)
5 Blank
6-9 The first data word
10 Blank
11-14 The second data word
. .
75 Blank
76-79 The fifteenth data word
```

BDPIP - 2311 BOM Disk Pack Initialization Program

82 1800 MPX Operating Procedures
The format of *GETALT is:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>Drive number of the 2311. This column must not be blank.</td>
</tr>
<tr>
<td>CCCT</td>
<td>Cylinder and track number (both in decimal) for which an alternate is requested. No alternate track can be assigned for track 0, cylinder 0.</td>
</tr>
<tr>
<td>ZZZZZZ</td>
<td>Existing volume serial number to be checked. If this field is blank, the job continues without checking. If a volume serial number is specified and does not agree with the volume serial number on the 1316 pack, the job is aborted.</td>
</tr>
</tbody>
</table>

The format of *INITLZ is:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>Drive number of the 2311 to be initialized.</td>
</tr>
<tr>
<td>CCCT</td>
<td>Cylinder and track number (both in decimal) where the Volume Table of Contents (VTOC) is to begin. The VTOC cannot begin at track 0 of cylinder 0 or at any alternate track.</td>
</tr>
<tr>
<td>MMMM</td>
<td>Length of the VTOC in tracks. This field must not be zero. This number can range from 0001 through 1990 (decimal).</td>
</tr>
<tr>
<td>YYYYYY</td>
<td>Volume serial number. This field can contain any alphabetic or numeric characters.</td>
</tr>
<tr>
<td>ZZZZZZ</td>
<td>Existing volume serial number to be checked. If this field is blank, initialization continues without checking. If a volume serial number is specified and is not equal to the volume serial number on the 1316 pack, the initialization job is aborted.</td>
</tr>
</tbody>
</table>
OPERATING PROCEDURES

1. Load BOM into main storage.

2. Place BDPIP into the card read punch hopper followed by 2311 initialization control cards (*END BDPIP must be the last control card) and a blank card.

3. Execute the BOM Absolute Loader (sense switch 0 ON, data switch 15 ON).

4. The following message is printed when program execution begins:

   BOM DISK PACK INITIALIZATION PROGRAM

5. Press console START.

6. At this point the first initialization control statement (*INITLZ or *GETALT) is read and the following message is printed:

   DRIVE SELECTED IS XX, IF CORRECT PUSH START WITH SENSE SWITCH 0 ON.

7. If the drive number is correct, press console START with sense switch 0 ON. If the drive is incorrect:

   1. Press START with sense switch 0 OFF; the program will return to its beginning (step 4).

   2. Then correct the drive number in the first initialization control statement and push console START. The program will return to step 6 and print the message again.

8. Press console STOP, RESET, and START to end the BDPIP operation.

When the 2311 initialization job has been completed, one of the following messages is printed:

1. If no defective tracks were found:

   THERE WERE NO DEFECTIVE TRACKS FOUND ON THIS PACK

2. If defective tracks were found:

   THE FOLLOWING TRACKS WERE FOUND DEFECTIVE ON THIS PACK:
   DEF TRACK ADDR	ALT TRACK ADDR
   CCCT	CCCT

   where the CCCT fields are the cylinder and track addresses of the defective and alternate tracks.

If program detects a control statement error, it will print an error message and execute a WAIT. When this happens, place the corrected control cards into the hopper, ready the read punch and press console START.

Note: To make your 1316 pack DOS-compatible, it must be initialized under DOS. BDPIP will initialize your 1316 to be OS-compatible.

BDUPL - BOM Disk Duplication Program

The BOM Disk Duplication Program copies the contents of an 1810 or 2311 drive onto another 1810 or 2311 drive. An 1810 mapped or physical drive can be copied onto another 1810 mapped or physical drive, while a 2311 drive can only be copied onto another 2311 drive. The drive being copied is called the source drive and the drive being copied onto is called the object drive.
OPERATING PROCEDURES

1. Load BOM into main storage.

2. Place BDUPL into the card read punch hopper followed by one blank card.

3. Execute the BOM Absolute Loader (sense switch 0 ON, data switch 15 ON).

4. The following message is printed when program execution begins.

BOM DISK COPY PROGRAM

SET DATA SWS TO SOURCE AND OBJECT
DRIVE CODES--XX0
SET DATA SW 1 ON IF 2311 COPY

5. Set the source drive number in data switches 4-7 and set the object drive number in data switches 12-15. Set data entry switch 1 ON if you are copying a 2311 drive.

6. Press console START.

7. The following message is printed if you are copying an 1810 drive:

SOURCE DRIVE IS 000M
OBJECT DRIVE IS 000N
IF CORRECT PUSH START WITH DATA SW 0 OFF

where M and N are the disk drive numbers (0-7).

The following message is printed if you are copying a 2311 drive:

2311 DISK COPY
SOURCE DRIVE IS 000M
OBJECT DRIVE IS 000N
IF CORRECT PUSH START WITH DATA SW 0 OFF

where M and N are the disk drive numbers (0-7).

8. M is the source drive number and N is the object drive number. If either of the drive numbers is in error, set data switch 0 ON and press console START to return to step 4. If both numbers are correct, press console START with data entry switch 0 OFF.

When the duplication is finished, the program returns to step 4. Press console STOP, RESET, and START to end the BDUPL operation.

Note that mapped 1810 drives can vary in size. If either the source or object drive, or both of them, is a mapped 1810 drive and they differ in sector size, the following message will be printed as you begin to execute the program:

SOURCE SECTORS XXXX
OBJECT SECTORS YYYY
PUSH START WITH SEN SW 1 ON TO EXECUTE

where:

XXXX is the number of sectors on the source drive.
YYYY is the number of sectors on the object drive.

If you decide to continue with the copy as specified, press console START with sense switch 1 ON. The number of sectors copied will be the smaller of XXXX and YYYY. If you wish to terminate the job, press console START with sense switch 1 OFF and the program will return to step 4.

The main purpose of copying drives of unequal size is to temporarily store the contents of a smaller drive onto a larger one. If you decide to continue copying, you will be unable to use the object drive as a system drive.
BDWAP - BOM Disk Write Addresses Program

The BOM Disk Write Addresses Program checks the disk surface of a 2315 cartridge and writes logical sector addresses in the sectors of those cylinders that are not defective. If a cylinder on the 2315 disk cartridge is damaged so that it is impossible to write on it, BDWAP will ignore that cylinder.

BDWAP must be used to initialize all new 2315 cartridges that are to be used with MPX. It may also be used at any other time that you wish to zero your 2315 cartridge and write addresses on it.

OPERATING PROCEDURES

1. Load BOM into main storage.
2. Place BDWAP into the card read punch hopper followed by a blank card.
3. Execute the BOM Absolute Loader (sense switch 0 ON, data switch 15 ON).
4. The following message is printed when program execution begins:

```
BOM DISK WRITE ADDRESSES PROGRAM
ENTER NO. TRIES ON DATA SW MAX 001F
```

5. Set the number of tries in the data entry switches, right-justified. The entry must be a hexadecimal number in the range /0001 through /001F.

The number of tries designates the number of times BDWAP will try to write the hexadecimal pattern /E5E5/1313 in every word of the 1810 drive.

6. Press console START.
7. If the number of tries entered is not acceptable, the following message is printed:

```
ENTER NO. TRIES ON DATA SW MAX 001F
```

8. Correct the entry and press console START. If the number of tries entered is acceptable, the following message is printed:

```
DATA SWITCHES EQUAL LOGICAL DRIVE
DRIVE CODES - HEX 0000 0001 0002
```

9. Set the drive number in the data switches as follows:

```
All off - drive 0
Switch 15 ON - drive 1
Switch 14 ON - drive 2
```

10. Press console START.

The drive number is checked for legality. If the drive number selected is not a legal drive number, the program returns to step 8. If the drive number is a legal drive number, the following message is printed:

```
DRIVE SELECTED IS X, IF CORRECT PUSH START WITH SENSE SWITCH 0 ON.
```
11. If the drive selection is correct, turn sense switch 0 ON and press console START. If the drive selection is incorrect, turn sense switch 0 OFF and press console START to return to step 8.

Addresses are now written on the 1810 drive and one of the following messages is printed:

a. THERE ARE NO DEFECTIVE CYLINDERS

Upon completion of this printout the program returns to step 4 and is ready to address another 2315 cartridge. To end the BDWAP operation, press console STOP, RESET, and START with all sense switches OFF.

b. If there are one, two, or three defective cylinders, the following message is printed:

CYLINDERS 00XX 00XX 00XX ARE DEFECTIVE

where the 00XX fields are the physical cylinder numbers (in hexadecimal) of the defective cylinders. The 2315 cartridge can be used. The program returns to step 4 and is ready to address another 2315 cartridge. To end the BDWAP operation, press console STOP, RESET, and START with all sense switches OFF.

c. If cylinder 0 is defective or if there are more than three defective cylinders, the following messages are printed:

THIS DISK CARTRIDGE IS NOT ACCEPTABLE TO MPX
TOO MANY CYLINDERS ARE BAD OR CYLINDER ZERO IS BAD
CYLINDERS (0000 OR 00XX ... 00XX) ARE DEFECTIVE

where the 00XX fields are the cylinder numbers (in hexadecimal) of the defective cylinders. The 2315 disk cartridge cannot be used with MPX. The program returns to step 4 to allow you to load a new 2315 disk cartridge.

To end the BDWAP operation, press console STOP, RESET, and START with all sense switches OFF.

d. If repeated seek failures occur, the following message is printed:

CANNOT COMPLETE SEEK-ABORT JOB

The program returns to step 4 to allow you to retry. To end the BDWAP operation press console STOP, RESET, and START with all sense switches OFF.

The physical cylinder numbers (Pc) of the defective cylinders are printed in hexadecimal. For each defective cylinder, the number stored in the Defective Cylinder Table on the disk is the logical sector number that would have been assigned to the first sector in the cylinder, had the cylinder not been defective. To compute the logical sector number (Ls) of the first sector of a defective cylinder, use this formula:

\[ L_s = (P_c \times 8) - (N \times 8) \]

Pc was described in the preceding paragraph, and N is the number of defective cylinders preceding the cylinder for which Ls is to be computed.

The cylinder numbers range from 0 through 202 in decimal, or 0 through CA in hexadecimal.

If, for example, you want to find the logical sector number of the first sector of cylinder 10, and cylinder 10 is the third defective cylinder on the cartridge, you can use the following calculations:

\[
L_s = (P_c \times 8) - (N \times 8) = 10 \times 8 - 2 \times 8 = 80 - 16 = 64
\]
BLIST – BOM 80-80 List and Sequence Check Program

The 80-80 List and Sequence Check Program in card form (BLIST), is one of the card utilities punched during system generation. The 80-80 List and Sequence Check Program lists card statements on the list printer and/or checks the progression of sequence numbers in card columns 73-80. If, during a sequence check, a card containing a sequence number lower than the preceding one, or containing blanks, is found, an error message is printed and the system executes a WAIT.

OPERATING PROCEDURES

1. Load BOM into main storage.
2. Place BLIST into the card read punch hopper followed by the cards to be listed and/or sequence checked and a blank card.
3. Execute the BOM Absolute Loader (sense switch 0 ON, data switch 15 ON).
4. The following message is printed when program execution begins:
   
   BOM 80/80 LIST--SEQUENCE CHECK  
   SEN SW 0 ON LIST CARDS  
   SEN SW 1 ON SEQUENCE CHECK

5. Set sense switch 0 ON if you wish to list cards and set sense switch 1 ON if you wish to perform a sequence check. Both switches may be ON. The switch settings may be changed while the program is in execution.
6. Press console START.

To terminate the program and return to BOM, turn OFF all sense switches and press console STOP, RESET, and START, in that order.

BRELD – BOM 1810 Disk Dump Reload Program

This program reloads the data punched into cards by the BOM Disk Dump to Cards Program (BDUMP) onto 1810 drive 0 (physical or mapped). The data is loaded in the same relative sectors it occupied on the original 1810 drive 0. If 1810 drive 0 is on a 2315 disk cartridge, the cartridge must have been previously initialized by the BOM Disk Write Addresses Program (BDWAP).

OPERATING PROCEDURES

1. Load BOM into main storage.
2. Place BRELD into the card read punch hopper followed by the output from the BOM Disk Dump to Cards Program (BDUMP) and two blank cards.
3. Execute the BOM Absolute Loader (sense switch 0 ON, sense switch 15 ON).
4. The following message is printed when program execution begins:
   
   BOM RELOAD OF DISK TO CARD DUMP

5. Press console START.
The program proceeds to load the data from cards to disk. When the End-of-All-Cylinders card is read, the program returns to step 4. To terminate the BRELD operation, press console STOP, RESET, and START.

A disk error during the reload process causes the job to be aborted. The program returns to step 4.

If the source or object drive is mapped on a 1316 disk pack and the drive you are loading is smaller than the drive that was dumped, all cards are loaded up to the addressed cylinder break card and the following message is printed:

BOM RELOAD OF DISK FROM CARDS
SRC SAD 640 EXCEEDS OBJ SAD JOB ABORTED
BOM RELOAD OF DISK FROM CARDS
MPX Debugging Aids

MPX includes programs to aid you in real-time and batch-processing system debugging. These comprise the following:

- BOM Utility Package
- Disk Pack Initialization Program
- Online Dump Program
- 80-80 List and Sequence Check Program
- 1442 Relocatable Card Dump Program
- Card Dump Analysis Program
- TRAP Subroutines
- CE Coreload Programs
- 1442 Card Reader Diagnostic

BOM Utility Package

This package contains a utility monitor (comprising four basic functions) and three trace subroutines (Full Trace, Check/Stop Trace, and Monitor Trace) which may be selectively included in the system at BOM assembly time via BOM equate cards.

UTILITY MONITOR

The Utility Monitor provides four basic functions: BOM or Executive Reload, Disk Dump, Dump, Main-Storage Dump, and Table Dump. Any of these functions can be executed in the following manner:

1. Press STOP and RESET on the 1800 console.

2. Set sense switches 0 and 1 for the function required according to the following table:

<table>
<thead>
<tr>
<th>Sense Switch 0</th>
<th>Sense Switch 1</th>
<th>Function Called</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>BOM or Executive Reload</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>Disk Dump</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>Main-Storage Dump</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>Table Dump for Check/Stop Trace</td>
</tr>
</tbody>
</table>

3. Press console START.

4. Follow the individual directions for the function called.

BOM OR EXECUTIVE RELOAD

You can force BOM or the Executive to perform a reload by pressing console STOP, RESET, and START with sense switches 0 and 1 OFF.
**BOM RELOAD**

When this function is called, the following standard BOM sense-switch-option messages are printed; BOM then proceeds to execute the reload function for a batch-processing system.

```
IBM 1800 MPX/BOM XX.XXX YY MTH ZZ
SEN SW 0 ON ABSOLUTE 'LOADER
SEN SW 1 ON BP MONITOR
SEN SW 2 ON SET CLOCK VIA DATA SWS
SEN SW 3 ON SET DATE VIA DATA SWS
```

Calling the BOM reload function allows you to return to the initial starting point of BOM without the necessity of reloading the BOM deck, or performing a cold start operation. You should also note that whenever batch processing is in progress under BOM or background processing, the job in progress can be aborted by turning program switch 7 ON and pressing CONSOLE INTERRUPT. This action causes the Batch-Processing Monitor Supervisor to be read into main storage. The Supervisor then starts searching the job stream for a // JOB statement.

**EXECUTIVE RELOAD**

With the Executive in direct control of a real-time system, you have the capability of initiating a system reload from the operator's console as indicated above. The action taken in a system reload is as follows:

1. Reread from disk the user-included subroutine area if this option was elected at BOM assembly.

2. Reread from disk the SPAR coreloads currently in main storage (as indicated by its word count and sector address in words 1 and 2 of the SPAR areas) if this option was elected at BOM assembly.

3. The system is now unmasked to the user-specified mask condition. If a reload coreload was specified at cold start, the system will reload this coreload and branch to it. If no reload coreload was specified, the system will perform a CALL LEVEL (to the level and bit specified on the BOM equates LDLEV and LDBIT) and a CALL EXIT on the basic level.

**DISK DUMP**

Calling the disk dump function allows you to dump specified portions of a disk on the list printer. The following printout appears when this function is called:

```
SEN SW 1 OFF FOR 2311 DISK DUMP
```

**1810 DUMP**

After the above printout, the following procedure is required:

1. Push console START with sense switch 1 ON.

2. This message is printed:

```
DATA SW = DRV CODE AND SEC ADD DAAA
```

3. Set sense switch 0 ON to dump two or more sectors. Leave sense switch 0 OFF to dump one sector.
4. For disk dumps, set sense switches 2 and 3 to indicate the character code required for disk analysis, as follows:

<table>
<thead>
<tr>
<th>Sense Switch 2</th>
<th>Sense Switch 3</th>
<th>Character Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>1053 Printer Code</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>1443 Printer Code</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>EBCDIC Code</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>Suppress character output</td>
</tr>
</tbody>
</table>

5. Enter the logical disk drive number (right-justified) in hexadecimal in data switches 0 through 3, and enter the hexadecimal starting sector address in data switches 4 through 15.

As long as sense switch 0 remains ON, sequential sectors starting at the starting sector address are dumped. When sense switch 0 is turned OFF, the sector being dumped is completed and the subroutine returns to step 2.

The format of the disk dump is as follows:

```
SECTOR XXXX FILE PROTECT ON/OFF
RELATIVE DATA ADD YYYY TO SECTOR ZZZZ
DISK READ ERROR PARAMETER 000E
```

```
QAAA DDDD DDDD DDDD ---- DDDD *ABCDE-----*
```

where:

- XXXX is the sector address.
- ON/OFF either ON or OFF is printed depending on the file protect status of the sector.
- YYYYY is the address of the first data word of the sector (relative to the first data word of the first sector dumped).
- ZZZZZ is the address of the first sector dumped.
- 000E is the completion code returned by BULKN following the read of the sector to be dumped. A correct read is indicated by a completion code of 1. If a completion code other than 1 is printed, check the completion code meaning to determine the validity of the data. (The contents of the read buffer are dumped independently of the completion code returned by BULKN.)
The BULKN completion codes are as follows:

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Successful completion of call</td>
</tr>
<tr>
<td>2</td>
<td>Device logically offline</td>
</tr>
<tr>
<td>3</td>
<td>Device not ready*</td>
</tr>
<tr>
<td>4</td>
<td>Parity error</td>
</tr>
<tr>
<td>5</td>
<td>Write select*</td>
</tr>
<tr>
<td>6</td>
<td>Data error</td>
</tr>
<tr>
<td>7</td>
<td>Data overrun</td>
</tr>
<tr>
<td>8</td>
<td>Seek error**</td>
</tr>
<tr>
<td>9</td>
<td>File protect error</td>
</tr>
<tr>
<td>A</td>
<td>Bad sector address on disk*</td>
</tr>
<tr>
<td>B</td>
<td>Address modification</td>
</tr>
</tbody>
</table>

*When one of these errors occur, the drive is automatically taken off line

**If the disk is a fast-access device (1810 Model B), it is taken off line when this error occurs

0AAA is the relative address of the next data word within the sector.

DDDD are the data words in the sector. Sixteen data words are printed per line on the 1443 printer; 8 data words per line on the 1053 printer.

*ABCDE---* is an analysis of the data words based on a specified character code selected under step 4. Periods are printed for both periods and unrecognizable codes. Two characters per word are printed.

If the contents of two or more succeeding lines of the dump are identical (for example, all zeros), only the first is printed. Printing is then suppressed until new data is found. The printer then takes an additional line space and prints the new line. The output is in hexadecimal.

Note: A disk dump uses a large part of VCORE as a buffer. The buffer begins at the higher-numbered addresses of VCORE; therefore, if you want a main-storage dump of this part of VCORE, you should get it prior to doing a disk dump operation which will destroy the contents of the area.
2311 DISK DUMP

The system prints out the following message:

SEN SW 1 OFF FOR 2311 DISK DUMP

Then carry out the following procedures:

1. Set sense switch 1 OFF.
2. Press console START.
3. The following message is printed:

2311 DATA SW=DRV-CYL-TRACK ADD DCCT

4. Set sense switch 0 ON to dump two or more records. Leave sense switch 0 OFF to dump one record.
5. For disk dumps, set sense switches 2 and 3 to indicate the character code required for disk analysis, as follows:

<table>
<thead>
<tr>
<th>Sense Switch 2</th>
<th>Sense Switch 3</th>
<th>Character Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>1053 Printer Code</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>1443 Printer Code</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>EBCDIC Code</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>Suppress character output</td>
</tr>
</tbody>
</table>

Note that the character code, not the output device, is selected at this time.
6. Enter the disk drive number (0,1,2,3,4,5,6 or 7) (right-justified) in hexadecimal in data switches 0 through 3 and enter the hexadecimal starting cylinder and track addresses in data switches 4 through 15.
7. Press console START.

As long as sense switch 0 remains ON, sequential records, starting at the starting disk address, are dumped. When sense switch 0 is turned OFF, one more record is dumped, and the subroutine returns to step 3.

The format of the disk dump is as follows:

```
DRIVE X CYLINDER YY  TRACK ZZ
ERROR PARAMETER 000E
RECORD WW
COUNT
AAAA CCCC CCCC CCCC ....
KEY BBBB KKKK KKKK KKKK ....
DATA
EEEE DDDD DDDD DDDD ....
```

where

- **X** is the 2311 drive number.
- **YY** is the cylinder being dumped.
- **ZZ** is the track being dumped.
- **000E** is the FILEN error parameter. The record is printed only for completion codes of 1 (successful I/O operation) and 6 (data check).
WW is the relative record on track ZZ.

AAAA is the relative count address.

CCCC is the count field.

BBBB is the relative key address.

KKKK is the key field.

EEEEE is the relative data word address.

DDDD is the data field.

Note: If an attempt is made to dump a 2311 record which is larger than VCORE, the error message RECORD EXCEEDS VCORE will be printed. Recovery is automatic. When subsequent records that fit into VCORE are read, they will be dumped. 2311 Disk Dump uses a large part of VCORE as a buffer. The buffer begins at the start of VCORE. Therefore, if you want a dump of VCORE, you should get it prior to doing a disk dump operation which will destroy the contents of the area.

MAIN-STORAGE DUMP

Calling the main-storage dump function allows you to dump selected portions of main storage on the list printer. The function can be called directly from your program or via the Utility Monitor sense switch options. The contents of the A-register and the Q-register are destroyed during a dump. When the dump is called via the Utility Monitor, the following procedure is required for execution:

1. Set sense switches 2 and 3 to indicate the character code required for main-storage interpretation, as follows:

<table>
<thead>
<tr>
<th>Sense Switch 2</th>
<th>Sense Switch 3</th>
<th>Character Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>1053 Printer Code</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>1443 Printer Code</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>EBCDIC Code</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>Suppress character output</td>
</tr>
</tbody>
</table>

2. Enter the hexadecimal starting address in the data switches.

3. Press console START.

4. Enter hexadecimal stopping address in the data switches.

5. Press console START.

6. When the main storage dump is completed, the system returns to step 1. A new dump can be initiated if desired. The format of the main-storage dump is as follows:

```
ADDR 0000 1111 2222 3333 .... .... 7777(1053) +ABCD----+
       FFFF(1443)
```

where:

ADDR is the starting main-storage address of the line of data words printed.

0000 is the contents of the ADDR+0 word of main storage.

1111 is the contents of the ADDR+1 word of main storage.
is the contents of the ADDR+2 word of main storage.

*ABCDE----*  is the interpretation of the data words on the basis of a specified character code (2 characters/word) selected in step 1 above.

Identical lines of printing are suppressed as in disk dump.

**TABLE DUMP (FOR CHECK/STOP TRACE)**

Calling the table dump function allows you to dump the addresses of the last ten executed instructions in a program. These addresses are stored in a table when the program is run in the trace mode; this table is updated by the Check/Stop Trace subroutine. Thus, if the trace mode is not used (console mode switch set to RUN), the table will contain no entries. The console mode switch must be set to RUN when the table is dumped.

On a table dump for Check/Stop Trace, the format is as follows:

```
TABLE DUMP ADDR-1 ADDR-2 ..............ADDR-10
SS AAAA QQQQ X1X1 X2X2 X3X3
```

where:

<table>
<thead>
<tr>
<th>ADDR-1</th>
<th>is the address of the last instruction executed on the basic level.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDR-2</td>
<td>is the address of the next to last instruction executed on the basic level.</td>
</tr>
<tr>
<td>ADDR-10</td>
<td>is the address of the tenth from the last instruction executed on the basic level.</td>
</tr>
<tr>
<td>SS</td>
<td>is the status of the carry (first digit) and overflow (second digit) indicators before the execution of the last instruction.</td>
</tr>
<tr>
<td>AAAA</td>
<td>is the contents of the A-register before the execution of the last instruction.</td>
</tr>
<tr>
<td>QQQQ</td>
<td>is the contents of the Q-register before the execution of the last instruction.</td>
</tr>
<tr>
<td>X1X1</td>
<td>is the contents of index register 1 before the execution of the last instruction.</td>
</tr>
<tr>
<td>X2X2</td>
<td>is the contents of index register 2 before the execution of the last instruction.</td>
</tr>
<tr>
<td>X3X3</td>
<td>is the contents of index register 3 before the execution of the last instruction.</td>
</tr>
</tbody>
</table>

**TRACE SUBROUTINES**

The trace subroutines are internal to the IBM system and may be set up at any time via CONSOLE INTERRUPT. The subroutines are used to trace a given range of addresses (Full Trace), monitor an address for change (Monitor Trace), or perform a table dump if the instruction in a selected address is executed (Check/Stop Trace).

The setup and execution of the trace subroutines are independent; that is, the trace parameters may be entered at any time and the actual trace performed at a later time when the program is executed. However, only one trace option may be used at a time.
Only basic-level coreloads may be traced.

**TRACE DEFINE LIMIT SUBROUTINE**

The Trace Define Limit subroutine allows you to select any one of the three trace subroutines. To call the Trace Define Limit subroutine, set program switch 5 ON, program switch 7 OFF, and press CONSOLE INTERRUPT.

Pressing CONSOLE INTERRUPT causes the following message to be printed (unless otherwise indicated, each message is followed by a WAIT):

**DATA SW0 ON FOR FULL TRACE**

If a full trace is desired, set data switch 0 ON, press console START, and follow the instructions for "Full Trace."

If you do not want a full trace, leave data switch 0 OFF and press console START. The following message is printed:

**DATA SW0 ON FOR MONITOR TRACE**

If a monitor trace is desired, set data switch 0 ON, press console START, and follow the instructions for "Monitor Trace."

If you do not want a monitor trace, leave data switch 0 OFF and press console START. The following message is printed:

**ADDR CHECK TRACE**

**DATA SW5 = CHECK ADDR AAAA**

You should now follow the instructions for "Check/Stop Trace."

Note: During the time the Trace Define Limit subroutine is in use, the MPX system is in a suspended state, that is, all levels are masked and the real-time clock is stopped. (If the clock was not stopped, the programmed timers might run out and all be waiting to interrupt as soon as the trace limits were entered and the levels were unmasked.

**FULL TRACE**

When you select the full trace option, the following message is printed:

**DATA SW5 = LOW TRACE ADDR LLLL**

Enter the low limit boundary of the trace in the data switches in hexadecimal, four switches per character, and press console START. The low limit of the trace (LLLL) is printed, followed by the following message:

**DATA SW5 = HIGH TRACE ADDR HHHH**

Enter the upper limit boundary of the trace in the data switches in hexadecimal. If the trace is desired immediately, that is, the program to be traced is currently suspended from execution, set the console mode switch to TRACE, turn program switch 5 OFF, ensure that data switch 15 is OFF (it can be turned OFF just after pressing console START if data switch 15 is used for address), and press console START. If the program is to be traced at a later time, leave the console mode switch in RUN, turn program switch 5 OFF, and press console START.

The upper limit of the trace (HHHH) is printed, followed by the message:

**DATA SW 15 ON SUPPRESSES PRINT**
Without waiting, the Trace Define Limit subroutine returns control to the program that was in execution when CONSOLE INTERRUPT was pressed.

Notes:

1. The trace printout can be suppressed at any time by turning data switch 15 ON. To return the system to normal operation, press console STOP, set the console mode switch to RUN, and press console START.

2. A full trace should not be used with a FORTRAN or assembler language program containing a PAUSE statement. If a PAUSE is encountered while printing in the TRACE mode, the program will loop on itself.

The full trace format is as follows:

ADDR  | INS1 | INS2 | AAAA | QQQQ | SS  | X1X1 | X2X2 | X3X3

where:

ADDR is the address of the next instruction to be executed.

INS1 is the first word of the next instruction to be executed.

INS2 is the second word of the next instruction to be executed.

AAAA is the contents of the accumulator before the execution of the next instruction.

QQQQ is the contents of the Q-register before the execution of the next instruction.

SS is the status of the carry (first digit) and overflow (second digit) indicators before the execution of the next instruction.

X1X1 is the contents of index register 1 before the execution of the next instruction.

X2X2 is the contents of index register 2 before the execution of the next instruction.

X3X3 is the contents of index register 3 before the execution of the next instruction.

A sample output of a full trace run is listed below:

F496  F500  F4D5  0400  0002  11  000B  FEB0  FEB3
F498  4C18  F4A0  0C00  0002  11  000B  FEB0  FEB3
F49A  C031  0C00  0002  11  000B  FEB0  FEB3
F49B  71FF  0400  0002  11  000B  FEB0  FEB3
F49C  70F9  0400  0002  11  000A  FEB0  FEB3
F496  F500  F4D5  0400  0002  11  000A  FEB0  FEB3
F498  4C18  F4A0  0000  0002  11  000A  FEB0  FEB3
F4A0  1090  0000  0002  11  000A  FEB0  FEB3

MONITOR TRACE

The Monitor Trace subroutine checks a specified word of main storage after every instruction and notifies you if that word has changed in value since the instruction was executed. Thus, each time the specified word changes in value, monitor trace prints the value of the I-register before the change, after the change, and the new value of the monitored word.
Following this, the subroutine continues execution of the basic-level program and checks for a subsequent change in the value of the monitored word.

When the monitor trace option is selected, the following message is printed:

`DATA SWS = MONITOR ADDR`

Enter the address that is to be monitored for a change in the data switches in hexadecimal, four switches per character, turn program switch 5 OFF, and press console START (if an immediate trace is desired, set the console mode switch to TRACE before pressing console START).

The following message is printed:

`MONITOR CHECK ADDRESS IS CCCC`

Without waiting, the Trace Define Limit subroutine returns control to the program that was in execution when CONSOLE INTERRUPT was pressed.

The monitor trace format is as follows:

`AAAA BBBB CCCC`

- `AAAA` is the value of the I-register after the word changed.
- `BBBB` is the value of the I-register before the word changed.
- `CCCC` is the new value of the word.

**CHECK/STOP TRACE**

The Check/Stop Trace subroutine performs two functions, as explained below. The first function is performed only if data switch 15 is OFF during program execution.

1. If, during the execution of any basic-level program, the address of the next instruction to be executed is equal to the check/stop address, a table dump is performed followed by a WAIT. To continue execution, press console START. If a WAIT occurs without a table dump, the next instruction in the program being traced is a WAIT, and it may not reference the check/stop address. To continue, press console START. Note that if the check/stop address is branched to as a result of a BSI, or follows an XIO, the Check/Stop subroutine will not wait when the address is reached. BSI and XIO instructions cause the trace level to be masked for one instruction, thus missing the stopping address.

2. A table of the addresses of the last ten instructions that have been executed is maintained by check/stop trace. This table is dumped via the trace define limits subroutine. The mode switch must be set to TRACE when the address table is dumped.

When the check/stop trace option is selected, the following message will have already been printed:

`DATA SW = CHECK ADDR AAAA`

Enter the address to be checked in the data switches in hexadecimal (four switches per character), turn program switch 5 OFF and press console START. (If an immediate trace with both options is required, ensure that data switch 15 is OFF and set console mode switch to TRACE before pressing console START. Data switch 15 can be turned OFF just after pressing console START if it is used for the address.) The check address (AAAA) is printed followed by the following message:

`DATA SW 15 ON SUPPRESSES ADDR CHECK`
Without waiting, the Trace Define Limit subroutine returns control to the program that was in execution when CONSOLE INTERRUPT was pressed.

**DISK PACK INITIALIZATION PROGRAM**

The Disk Pack Initialization Program (DPIP) is a program in the Subroutine Library. You can build it into a batch-processing or process coreload and use it to initialize your 1316 disk packs.

DPIP has the same functions as BDPIP, which was described under "BOM Card Utilities." You can use DPIP by means of the following sequence of statements in the input stream:

```
1-10       11-20       21-30       31-40       41-50       51-60
1234567890 1234567890 1234567890 1234567890 1234567890 1234567890

1. /J JOB
2. /X EQ DPIP
3. **CCEND
4. **INITLZ and *GETALT statements
5. **
6. **
7. **END DPIP
8. **
```

You can also, of course, use *STORECI to store the coreload so that you don't have to build the coreload every time you want to use it.

**ONLINE DUMP PROGRAM**

The Online Dump Program (OLDMP) is a program in the Subroutine Library. You can build it into a process or batch-processing coreload and use it to dump selected parts of main storage, 1810 drives, and 2311 drives to the list printer. The size of the partition for which the coreload is built must be at least 3600 words.

To build OLDMP into a batch-processing coreload and execute it, use the following sequence of control statements:

```
1-10       11-20       21-30       31-40       41-50       51-60
1234567890 1234567890 1234567890 1234567890 1234567890 1234567890

1. /J JOB
2. /X EQ OLDMP
3. **CCEND
4. **
```

You can also store the coreload by using *STORECI and execute it in any way you choose.
At the beginning of its execution, OLDMP prints these messages:

ON-LINE DUMP PGM
SEN SW 0 CORE DUMP
SEN SW 1 DISK DUMP
SEN SW 2 TERMINATE

Set the sense switches as indicated in the messages and press console START.

The rest of the operating procedure and the format of the dump are identical to those described under "Disk Dump" and "Main-Storage Dump" in the section "BOM Utility Package." After the dump has been completed, the program prints the initial messages again, allowing you to carry out another dump operation. To stop execution of the program, at this point, when running under BOM, press console STOP, RESET, and START. To stop execution of the program, at this point, when running under the Executive (online), turn sense switch 2 ON and press console START.

80-80 LIST AND SEQUENCE CHECK PROGRAM

The 80-80 List and Sequence Check Program (MLIST) is a program in the Subroutine Library. You can build it into a batch-processing or process coreload and use it to list cards and/or to check their sequence numbers.

MLIST has the same functions as the BOM card utility BLIST. You can use MLIST by means of the following sequence of statements:

```
1-10
11-20
21-30
31-40
41-50
51-60
1
2
3
4
5
6
7
8

Cards to be processed by MLIST
```

You can also store the coreload by using *STORECI and execute it in any way you choose.

1442 RELOCATABLE CARD DUMP PROGRAM (CRDMP)

This is a stand-alone eight-card relocatable program that is used for real-time or batch-processing system debugging. It dumps all of main storage to cards in a special compressed binary format for analysis by the Card Dump Analysis Program (DMPAN). DMPAN then produces a formatted dump on the list printer.

The Relocatable Card Dump Program occupies approximately 340 main-storage locations and can be loaded into any even main-storage address.

Note: If the Relocatable Card Dump Program overlays the CDW table in a partition, DMPAN will suppress the printing of all labels in that partition.
The 1442 Relocatable Card Dump Program is operated in the following manner:

1. Press IMMEDIATE STOP or STOP and RESET on the console.

2. Clear storage protect bits by performing a clear main storage operation with the console mode switch set to DISPLAY.

3. Set the Check/Stop switch ON.

4. Set the console mode switch to LOAD.

5. Set the loading address for the CRDMP program in the data switches in hexadecimal (this must be an even storage address).

6. Press console RESET and LOAD I.

7. Place the eight-card dump program in the card read punch hopper followed by blank cards and press reader START. (Note that the program does not check for blank cards prior to punching. Input cards must therefore be blank or checksum errors will occur when they are read by DMPAN.)

8. Set the console mode switch to SI W/CS.

9. Press console PROGRAM LOAD.

10. After the first card has been read, set the console mode switch to LOAD.

11. Press console RESET and LOAD I (the data switches are still set to the load address).

12. Set the console mode switch to RUN and press START.

The cards punched by CRDMP are stacker selected. After all of main storage has been dumped, the program comes to a WAIT with /XXXX+7 displayed in the l-register. /XXXX is the CRDMP program load address. At this point the main-storage-to-card dump program can be repeated by pressing console START. To return to system operation, perform a cold start.

Note: This program can be loaded into location 0 using the standard program load procedure.

CARD FORMAT

Cards are punched in a compressed binary format that allows 56 data words per card. The card formats are as follows:

FIRST CARD

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/FFFF</td>
</tr>
<tr>
<td>3</td>
<td>Starting address of dump program</td>
</tr>
<tr>
<td>4</td>
<td>Ending address of dump program</td>
</tr>
</tbody>
</table>

DATA CARDS

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Beginning main-storage address for data on card</td>
</tr>
<tr>
<td>2</td>
<td>Checksum (sum of data words 3–60)</td>
</tr>
<tr>
<td>3</td>
<td>Word Count for data words on card</td>
</tr>
</tbody>
</table>
ERROR PROCEDURES

Punch errors during a dump cause the program to wait with /XXX+15 displayed in the I-register. /XXXX is the CRDMP program load address. Remove the card in error (last card in stacker) and press console START to repunch that card.

Read errors during the loading of the CRDMP program cause a dynamic WAIT. The loading process must be reinitialized (restart at step 1).

If the card read punch becomes not ready, the program loops until the reader is made ready.

Card Dump Analysis Program (DMPAN)

The Card Dump Analysis Program serves as a diagnostic aid to label all pertinent areas of main storage. It can be used in conjunction with the 1442 Relocatable Card Dump Program to obtain a formatted dump of main storage, in which case the card output of the CRDMP program is the input to DMPAN, or can be used directly to dump any portion of main storage. In either case, the requirements of the dump are dictated by parameters in the DMPAN control statement. DMPAN provides you with the ability to acquire a main-storage map of your real-time or batch-processing system. DMPAN is stored in the User Library on disk.

DMPAN is loaded into, and therefore destroys the contents of, the first 5.1K words of VCORE. If you want your analysis to reflect the original contents of this part of main storage, you should use the 1442 Relocatable Card Dump Program (CRDMP) and use the card deck it produces as input to DMPAN. Otherwise, you may use DMPAN directly. You specify your options in a DMPAN control statement. Figure 13 illustrates the stacked input required for DMPAN.

DMPAN can be executed in minimum VCORE (5140 words) provided the number of executive transfer vectors required by your system does not exceed 35 (105 words).
DMPAN CONTROL CARD FORMAT

The control card format for the Card Dump Analysis Program is shown in Figure 14.

<table>
<thead>
<tr>
<th>Column</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-4</td>
<td>Hexadecimal low main-storage address of area to be dumped. If no dumping is required, this parameter must be $\text{FFFF}$.</td>
</tr>
<tr>
<td>6-9</td>
<td>Hexadecimal high main-storage address of area to be dumped.</td>
</tr>
<tr>
<td>11</td>
<td>Blank</td>
</tr>
<tr>
<td></td>
<td>- Print all labels</td>
</tr>
<tr>
<td></td>
<td>- Print only those labels for the area to be dumped</td>
</tr>
<tr>
<td></td>
<td>- Do not print any labels</td>
</tr>
<tr>
<td>13</td>
<td>Blank</td>
</tr>
<tr>
<td></td>
<td>(or zero) - Dump line interpreted in 1053 printer code</td>
</tr>
<tr>
<td></td>
<td>1 - 1443 printer code</td>
</tr>
<tr>
<td></td>
<td>2 - EBCDIC code</td>
</tr>
<tr>
<td></td>
<td>3 - Suppress line code interpretation</td>
</tr>
<tr>
<td>15</td>
<td>Blank</td>
</tr>
<tr>
<td></td>
<td>(or zero) - Dump from cards</td>
</tr>
<tr>
<td></td>
<td>1 - Dump from main-storage</td>
</tr>
<tr>
<td>20-59</td>
<td>Any remarks. These remarks will appear in the page heading for each 1443 page.</td>
</tr>
</tbody>
</table>

Figure 14. DMPAN Control Card Format

Note: If columns 1-4 and 6-9 are all blank, all of main storage is dumped. Thus, a control card with a 1 in column 15 only gives a completely labeled dump of main storage on the list printer with each line interpreted in 1053 printer code.
The output of a dump analysis is a listing on the list printer of the specified main-storage dump which details all major Executive Director programs and tables, Executive I/O programs and tables, partitions and associated tables, and other areas of main storage as required, separately identified and labeled. Various control card options allow specification of start and end addresses of the area to be dumped, the type of printer character code required, and individual suppression of dumping and labeling. This program versatility enables you to request a main-storage map of the dumped system without a listing of the dumped area.

AREAS TO BE DUMPED

Depending on the particular system, the following programs, areas, and tables are labeled separately in the listing.

   Fixed Area
   Error Trap Area
   INSKEL COMMON
   All device tables
   The following Executive I/O subroutines:

   FILEN  IOSET  PUTQ
   BULKN  IOTST  GETQ
   TYPEN  RSAVE  EACRL
   PRNTN  TVSET  ABOERT
   CEINT  BNDSH  DISKZ
   IOEXT  IODRT  RINFO
   IOSAV  IOERR  INTB
   TVSAV  DIRCL  RSTDK
   TVEXT  STPRT  EACPT
   QZSAV  LINER  RELDO
   QZEXT  EREND  ILEVE
   BKSAV  PUTQO
   BKEXT  GETQO

   Error Message Table
   BOM Utility Programs
   *ICLT communications table
   *Disk save area
   *Queue table
   *Interval timer tables
   *Level coreload list control table
   *Empty queue list table
   *ICL Table
   *Partition tables
   *Program timer table
   *All level work areas
   *Real-time basic level work area
   *Background basic level work area
   *Interrupt branch table
   *Boundary table
   *Area busy table
   *Start of MIC code
   *I/O exit/entry in Executive Director
   *Queueing entry in Executive Director
   *CALL EXIT entry in Executive Director
   *Suspend level entry in Executive Director
   *CALL LINK entry in Executive Director
   *End-background-processing entry in Executive Director
   *FORTRAN I/O buffers in Executive Director
   *User-included subroutines
   *Executive CLNT
   *Executive FIO table
*Executive TV table
*Executive branch table
**Partitions
**VCORE

*These entries appear only for a real-time system.
**For each partition, the following tables, if they exist, are labeled:

<table>
<thead>
<tr>
<th>CDW</th>
<th>ETV</th>
<th>LPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIO</td>
<td>CLNT</td>
<td>DFT</td>
</tr>
</tbody>
</table>
| IST (only in SPAR partitions)

OPERATING PROCEDURES

The Card Dump Analysis Program is executed as a batch-processing relocatable program in VCORE under the control of the Batch-Processing Monitor in a real-time or batch-processing environment.

Depending on the state of the system at the time of an error, control of the Batch-Processing Monitor can be obtained in one of three ways.

1. Cold start to BOM.
2. Cold start to BPMON.
3. Press CONSOLE INTERRUPT with program switch 7 ON (in a real-time system).

Note: If the dump is directly from main storage, two blank cards must follow the control card. This notifies the system that the dump is from main storage instead of from cards.

OUTPUT FORMAT

The format of a dump line is identical to that of the dump format of the BOM Utility Monitor Dump Program, including the optional line interpretation into 1053 Printer, 1443 Printer, or EBCDIC code. The first and last lines of a labeled portion of the dump will list only those words that are within the area being dumped. The remaining words of these lines are filled with asterisks. This format simplifies recognition of the start address of a labeled portion. Each page of a 1443 printer dump will be numbered and can include a header containing your comments.

ERROR AND TERMINATION PROCEDURES

An invalid checksum (word 2) on an input card or a sequence number out of order initiates a terminal error procedure: an EACPT message is printed and the program waits. The card in error is the last card in the stacker. If data switch 15 is OFF when console START is pressed, the program continues to read cards from the point of error. This allows the card in error to be corrected or discarded. If data switch 15 is ON, the program reinitializes itself to read a completely new card deck. Cards with a sequence number of zero are not checksummed. Sequence numbers do not have to be consecutive.

SAMPLE DUMP ANALYSIS

An example of the use of the Card Dump Analysis Program is shown below. In this example, all of main storage was dumped, but only a representative sample of the output, interpreted in 1053 Printer Code, is included.
**IPX CORE DUMP ANALYSIS**

**OUTPUT ON 1443 FROM CORE 0000 - 0600**

| ADDR | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| B510 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0460 | 0500 | 0400 | FFFC | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 |
| 0520 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 |
| 0530 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 |
| 0540 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 |

**BULK00 DEVICE TABLE**

<table>
<thead>
<tr>
<th>0540</th>
<th>****</th>
<th>****</th>
<th>****</th>
<th>****</th>
<th>****</th>
<th>****</th>
<th>****</th>
<th>****</th>
<th>6600</th>
<th>0538</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
</tr>
</thead>
<tbody>
<tr>
<td>0550</td>
<td>0000</td>
<td>0000</td>
<td>0100</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
</tr>
<tr>
<td>0560</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
</tr>
<tr>
<td>0570</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
</tr>
<tr>
<td>0580</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
</tr>
<tr>
<td>0590</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
</tr>
</tbody>
</table>

**TYPE01 DEVICE TABLE**

<table>
<thead>
<tr>
<th>05C0</th>
<th>****</th>
<th>****</th>
<th>****</th>
<th>****</th>
<th>****</th>
<th>****</th>
<th>****</th>
<th>****</th>
<th>6600</th>
<th>0538</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
<th>0500</th>
</tr>
</thead>
<tbody>
<tr>
<td>05D0</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
</tr>
</tbody>
</table>

**FILEN SUBROUTINE**

| 0C3A | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** |

**BULKN SUBROUTINE**

| 0548 | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** |

**Card Dump Analysis Program (DMPAN) 109**
Optional Material Tape Dump Program

The Optional Material Tape Dump Program (MOPTP) provides you with the ability to punch, list, or punch and list selected modules from the optional material tape. MOPTP is stored in the User Area and is executed as background-processing job. The desired module and option are specified in control cards that you punch. Figure 14.1 illustrates the stacked input required for MOPTP.

MOPTP Control Card Format

The control card format for the Optional Material Tape Dump Program is shown in Figure 14.2

<table>
<thead>
<tr>
<th>Column</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-10</td>
<td>Name of the desired module. These begin with $S, and all unused positions must be filled with dashes (for example: $SASMASM--). An index of the modules is supplied with the optional material tape.</td>
</tr>
<tr>
<td>12</td>
<td>Output Option</td>
</tr>
<tr>
<td>L</td>
<td>List only</td>
</tr>
<tr>
<td>B</td>
<td>Punch and list</td>
</tr>
<tr>
<td>Blank or any other character</td>
<td>Punch only</td>
</tr>
</tbody>
</table>

Note: The last control card must contain $END in columns 1-4 MOPTP to exit.

Operating Procedures

The Optional Material Tape Dump Program is executed as a batch- or background-processing program in VCORE.

The following steps are required to execute the Optional Material Tape Dump Program:

1. Punch the required Supervisor control cards (Figure 14.1).
2. Punch the MOPTP control cards required to select the desired module and output option (Figure 14.2).
3. Place the optional material tape on either of the tape drives. The program will pause to allow specification of the drive selected.
4. Place the Supervisor and MOPTP control cards in the card reader (Figure 14.1) and ready the reader. Note that modules may be selected in any sequence but if modules are requested in any order other than that shown in the index, tape rewind may occur before processing of some of the modules.

5. Request execution of the Batch-Processing Monitor.

6. The control cards will be read. When MOPTP has begun execution, the following messages are printed on the list printer. (The program pauses with /0003 in the accumulator):

   SENSE SW 5 SELECTS TAPE DRIVE.
   OFF FOR DRIVE 0, ON FOR DRIVE 1.

Set sense switch 5 as required and press START. The MOPTP program will read the provided control cards and dump the selected modules as directed.

Output Format

The output of the Optional Material Tape Dump Program is punched cards, a listing, or both, in 1800 assembler source format. The cards are punched on 1442-0 and are acceptable as input to the 1800 assembler. Listings are printed on the list printer, which may be an 1816/1053 or a 1443.

Error and Termination Procedures

Termination of the Current Module

Turning on sense switch 3 while a module is being dumped will cause dumping of that module to be terminated and initiate reading of the next control card.

I/O Error Retry

All I/O errors will cause a pause with /0909 in the accumulator. Pressing START with sense switch 4 OFF will cause a retry of the I/O operation. In the case of a tape read error, a backspace will be performed first. Pressing START with sense switch 4 ON will cause the program to exit.

MOPTP Errors Detected

The following errors are detected by the MOPTP program. (All messages are printed on the list printer):

- INVALID CONTROL CARD
  A control card containing one of the following errors was read:

  Column 1 does not contain a $.
  Column 2 is neither S nor E.
  Columns 3-10 contain a blank.
  Column 11 is not blank.

  After printing the message, the program pauses with /0001 in the accumulator. It will continue reading control cards if START is pressed.

- NEED BLANK CARDS
  A nonblank card was detected during punching. The program pauses with /0002 in the accumulator. It retries the punch operation if START is pressed.

- UNABLE TO LOCATE $SXXXXXXXX
  The end of tape has been reached without locating the selected module. The program rewinds the tape and continues reading control cards.
Trap Subroutines

The trap subroutines are provided as real-time diagnostic aids that allow you to transfer the contents of a main-storage area to a buffer without stopping the system. If, in a real-time system, you do not want the system to stop when an EAC error occurs (sense switch 6 OFF), you may determine the cause of the error by using the trap subroutines.

Most programs do not have the built-in ability to retain error information. Thus, any information that could be obtained from a post-mortem dump would be of questionable value. It is therefore desirable to have a procedure that will give a dump of the affected area of main storage after the error occurs, but before the error information is lost. This objective is accomplished by "trapping" the error, and transferring the error information to a buffer. At some later time, this information can be dumped to an output device for closer analysis.

To achieve the data transfer, a CALL TRAP is made defining the error type, the error condition, and the name of the coreload in which the error occurs. The CALL TRAP statement can be issued from any coreload within your system. Once the error occurs, a CALL DTRAP can be executed from any coreload in the system to dump the area of the main storage specified in the CALL TRAP parameters on the system printer. The number of words trapped is dependent on the size of the trap buffer defined by the BOM equate cards TBUFS at system generation.

The TRAP and DTRAP subroutines reside in the User Library, while the trapping mechanism exists within EAC in the Executive I/O. The calling sequences for the trap subroutines are described in the Subroutine Library manual, Order Number GC26-3724.
The following example illustrates the use of the trap subroutines. Assume the following coreload, called SEEK1, has been built and is called for execution

// JOB 00.911 HRS
// ASM SEEK1
*LIST

0000 0 0818 START X10 DATA SENSE DATA SWS
0001 0 4C100000 BSC L START,- LOOP IF DATSW 0 IS OFF
0002 0 1001 SLA 1 IF ON--SHIFT OUT
0004 0 0000 DC 0
0005 0 1801 SRA 1 DATSW 0 AND
0006 0 D018 STO ARSA+1 STORE IN I/O AREA(SEC ADD)
0007 30 02913495 CALL BULKN SEEK TO
0009 1 0012 DC LIST SEC ADD
000A 0 C007 LD LIST TEST FOR
000B 0 4C20000A BSC L *-3,Z BUSY
000D 0 C80E X10 DATA SENSE DATA SWS
000E 0 4C280000 BSC L *-3,+Z LOOP IF DATSW 0 IS ON
0010 0 4C000000 BSC L START

**********************************************************
* BULKN I/O LIST *
**********************************************************

0012 0 0000 LIST DC 0 LINK/BUSY
0013 0 0000-- DC 0 TYPE-1 EXIT
0014 0 0004 BSS 4 SYSTEM RESERVED PARAM
0018 0 0000 DC 0
0019 0 5000 DC /5000 SEEK CONTROL PARAM
001A 1 001E DC AREA I/O AREA ADDRESS
001C 0 0001 DATA BSS E 1 I/OCC TO
001D 0 0740 DC /0740 SENSE DATA SWS
001E 0 0140 AREA DC 320 WD COUNT
001F 0 0000 DC *** SEC ADD
0020 0 0140 BSS 320
0160 0 0000 END START

NO ERRORS IN ABOVE ASSEMBLY.
SEEKI
DMP FUNCTION COMPLETED
// XEQ SEEK1 L
*CCEND

MPX, BUILD SEEK1

CORE LOAD MAP
TYPE NAME ARG1 ARG2
*CDW TABLE 61AA 0012
*F10 TABLE 618C 001E
*CNT TABLE 61DA 0004
MAIN SEEK1 61DE
CLNT SEEK1 61DC
CORE 6340 1CC0
MPX, SEEK1 LD XQ

As soon as the system attempts to execute the DC 0, the following op-code error occurs and the system performs a restart:

00.676 1-LEVEL OP-CODE ER SEEK1 FF18 800000 61E3 007F 344A 625A RS

The system restart effectively destroys any information that could be obtained by a main-storage dump. The error data must then be trapped. To trap the error data, write a CALL TRAP program to trap on the parameters listed in the EAC printout. The starting address of the program is found on the coreload map (in this case it's the address MAIN SEEK1 61DE). The trap program can be included in any coreload. When the trap program is loaded and executed, it sets up indicators in the system that cause the error data to be trapped when the error occurs.
The following program, called TRAP1, will trap the op-code error generated by the execution of SEEK1.

// JOB 00.899 HRS
// ASM TRAP1
*LIST

```
0000 30 236415C0 START CALL TRAP
0002  1 0014    DC LEVAR
0003  1 0009    DC MESS1
0004  1 000C    DC MESS2
0005  1 0011    DC MESS3
0006  0 61DE    DC /SIDE
0007 30 059C98C0 CALL EXIT
0009  0 0006    MESS1 DMES I-LEVL'E
000C  0 000A    MESS2 DMES OP-CODE ER'E
0011  0 0006    MESS3 DMES SEEK1 'E

0014  0 FF18    LEVAR DC /FF18
0016  0000    END START
```

NO ERRORS IN ABOVE ASSEMBLY.
TRAP1
DMP FUNCTION COMPLETED
// XEQ TRAP1 L
*CCEND

MPX, BUILD TRAP1

CORE LOAD MAP
TYPE NAME ARG1 ARG2
*CDW TABLE 61AA 0012
*F10 TABLE 61BC 001E
*CNT TABLE 61DA 0004
MAIN TRAP1 61DE
CLNT TRAP1 61DC
CALL TRAP 61FR R
CORE 6276 10BA
MPX, TRAP1 LD XQ

Now if SEEK1 is executed, the error data is stored in the trap buffer. To read the trap buffer write a program, in this case called TRAPD, to call DTRAP. This program can be included in any coreload; however, the most logical place is the restart coreload for the partition in which the error occurs.
The following program will dump the op-code error data for analysis. The output is on the system printer:

```
// JOB 00.925 HRS
// ASM TRAPD
*LST

0000 30 048D9057  JO CALL DTRAP
0002 30 059C98C0  CALL EXIT
0004 0000  END  JO

NO ERRORS IN ABOVE ASSEMBLY.

TRAPD
DMP FUNCTION COMPLETED
// XEQ TRAPD L
*CCEND

MPX, BUILD TRAPD

CORE LOAD MAP
TYPE NAME ARG1 ARG2
*CDW TABLE 61AA 0012
*F10 TABLE 618C 001E
*CNT TABLE 61DA 0004
MAIN TRAPD 61DE
CLNT TRAPD 61DC
CALL DTRAP 61E3
CORE 62B4 1D4C
MPX, TRAPD LD XQ

The system printer output is listed below:

```
ADDR ***0 ***1 ***2 ***3 ***4 ***5 ***6 ***7
61D8 **** **** **** **** 081B 4C10
61E0 61DE 1001 0000 1801 D018 4880 007C 61F0
61E8 0007 4C20 618E 080E 4C28 61EB 4000 61DE
61F0 0000 0000 0000 0000 0000 0000 0000 0000 5000
61F8 61FC 0000 0000 0740 0140 0000 0000 0000
6200 0000 0000 0000 0000 0000 0000 0000 0000
6208 0000 0000 0000 0000 0000 0000 0000 0000
6210 0000 0000 0000 0000 0000 0000 0000 0000
6218 0000 0000 0000 0000 0000 0000 0000 0000
6220 0000 0000 0000 0000 0000 0000 0000 0000
6228 0000 0000 0000 0000 0000 0000 0000 0000
6230 0000 0000 0000 0000 0000 0000 0000 0000
6238 0000 0000 0000 0000 0000 0000 0000 0000
6240 0000 0000 0000 0000 0000 0000 0000 0000
6248 0000 0000 0000 0000 0000 0000 0000 0000
```

For comparison purposes, the same area of main storage was dumped before program execution by the BOM Utility Monitor.

```
CE Coreload Programs (CECLD, CECLX)

Two CE Coreload programs, CECLD and CECLX, enable you to print out and modify the status of I/O devices on the system. CECLX includes extensions that can do 2311 disk storage drive, communications adapter, and 2790 data communication system functions additional to the functions of CECLD.

The CE Coreload programs are stored in the User Library on disk. CECLD requires approximately 1000 words of main storage and CECLX requires approximately 5000 words of main storage. The functions available with CECLD and CECLX are:

- Set On/Offline status
- Reset hardware error count
- Set logical and physical device assignments for disk storage units, 1053 printers, and communications adapters
- Read out and reset Executive Director error counts

A 2311 extension of CECLX enables you to do the following:

- Dump and reset the 2311 Error Statistics Table
- Dump and reset the CE Error Log

A communications adapter extension in CECLX enables you to do the following:

- Dump and reset the Error Statistics Table
- Dump and reset the CE Error Log
- Initialize the communications adapter line trace buffer
- Dump Communications adapter line trace buffer

A 2790 extension of CECLX enables you to do the following:

- Dump and reset the 2790 error counts
- Dump and reset the 2790 error log
- Start a 2790 loop segment

Initial entry to CECLX is to the basic program, but special CECLX branch functions allow transfer of control to:

- The 2311, 2790, or communications adapter extension, from either the basic program or the other extension
- The basic program, from either extension
- The Batch-Processing Monitor Supervisor, from either the basic program or its extensions

Both coreloads handle various kinds of errors by producing coded displays in the A-register accompanied by attempts to type a message on the system printer (unsuccessful attempts if the printer has been taken off line).
HOW TO BUILD CE CORELOADS

Each CE Coreload program may be built into any kind of coreload. In a real-time system, a CE Coreload program is usually executed as a process coreload, queued to service the programmed interrupt which the system generates when the CE INTERRUPT button is pressed. The interrupt level and ILSW bit associated with the CE INTERRUPT button are determined during BOM assembly by the CELVL and CEBIT equate cards.

If the CE Coreload is built to service an interrupt, the interrupt should probably be assigned to a low level so that it won't disrupt normal operation of the system.

In a batch-processing system, the CE Coreload is executed as a batch-processing coreload under control of the Batch-Processing Monitor.

HOW TO USE A CE CORELOAD

You specify CE Coreload functions using the CE sense switches, which are located on the CE panel directly below the Processor-Controller console. You initiate each function by pressing console START. The system executes a WAIT when each function is completed.

When execution of the CE Coreload begins, the following message is printed:

SET FUNC IN CE SWITCHES

Unless you intend to branch to communications adapter, 2311, or 2790 extensions of CECLX, your first function must be Select Device Type.

CE CORELOAD FUNCTIONS

The following functions can be done by both CE Coreloads (CECLD and CECLX):

SELECT DEVICE TYPE

1. Set CE sense switches to 0010 XXXX, where XXXX has one of the following values.

<table>
<thead>
<tr>
<th>Setting of XXXX</th>
<th>Selected Device</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>1810 disk storage unit</td>
</tr>
<tr>
<td>0001</td>
<td>1816/1053 printer</td>
</tr>
<tr>
<td>0010</td>
<td>1443 printer</td>
</tr>
<tr>
<td>0011</td>
<td>1442 card read punch</td>
</tr>
<tr>
<td>0100</td>
<td>2401 magnetic tape</td>
</tr>
<tr>
<td>0101</td>
<td>A1 - basic</td>
</tr>
<tr>
<td>0110</td>
<td>A1 - expander</td>
</tr>
<tr>
<td>0111</td>
<td>1054 paper tape reader</td>
</tr>
<tr>
<td>1000</td>
<td>1055 paper tape punch</td>
</tr>
<tr>
<td>1001</td>
<td>DI</td>
</tr>
<tr>
<td>1010</td>
<td>DAO</td>
</tr>
<tr>
<td>1011</td>
<td>1627 plotter</td>
</tr>
<tr>
<td>1100</td>
<td>communications adapter</td>
</tr>
<tr>
<td>1101</td>
<td>2311 disk storage drive</td>
</tr>
</tbody>
</table>
2. Press console START.

For each logical device of the unit type specified, the following items are typed out:

- Logical device number
- Physical unit identification
- On/Offline status
- Hardware error count, if applicable.
This page intentionally left blank.
SET ON/OFFLINE STATUS

1. Set CE sense switches to:
   
   0100 0YYY to take the unit off line
   0101 0YYY to put the unit on line

   where:
   
   YYY is the logical device number of the unit (for 1053s use the logical device number minus one).

2. Press console START.

   The new status of the logical unit is typed out.

RESET ERROR COUNT

1. Set CE sense switches to 0110 0YYY.

   where:
   
   YYY is the logical device number of the unit (for 1053s use the logical device number minus one).

2. Press console START.

   The new status of the logical unit is typed out. Note that you should reset the error count at system generation to ensure that it is initialized properly before you use your system.

SWITCH LOGICAL UNIT ASSIGNMENTS

This function is only applicable to 1810 disk storage units, 2311 disk storage drives, 1053/1816 printers, and communications adapters.

1. Set CE sense switches to 11 XXX YYY.

   where:
   
   XXX is the physical device number (in binary) to be assigned to the logical device number YYY (for 1053s use the physical device number minus one).

   YYY is the logical device number to be assigned to the physical device XXX (for 1053s use the logical device number minus one).

2. Press console START.

   The new status of the logical unit is typed out.

READ OUT OR RESET EXECUTIVE ERROR COUNTS

1. Set CE sense switches to 1010 000X.

   where:
   
   X = 0 to type out Executive error counts
   X = 1 to reset all error counts.
2. Press console START.

The error counts are not typed out for the reset function. Note that you should reset these error counts at system generation to ensure that they are set properly before you use your system. You can terminate any CE coreload dump function at any time by changing the setting of the CE switches.

EXIT FROM CORELOAD

1. Set CE sense switches to 0000 0000.
2. Press console START.

The following is printed where XXXX is the error count.

<table>
<thead>
<tr>
<th>LABEL in EX-DIR</th>
<th>Error Word</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CWTD</td>
<td>NON-PROCESS SAVE XXXX</td>
<td>BULKN detected an error writing VCORE to the NPSV.</td>
</tr>
<tr>
<td>CRTSP</td>
<td>NON-PROCESS READ XXXX</td>
<td>BULKN detected an error reading the NPSV area to VCORE.</td>
</tr>
<tr>
<td>RQCE</td>
<td>QUEUE AREA READ XXXX</td>
<td>BULKN detected an error while reading a queued coreload into a partition.</td>
</tr>
<tr>
<td>QFULL</td>
<td>QUEUE TABLE FULL XXXX</td>
<td>QLEVL returned to MIC an error parameter other than 1, indicating the coreload queue was full, or the entry was already in the queue.</td>
</tr>
<tr>
<td>EIRD</td>
<td>COMMON VCORE READ XXXX</td>
<td>BULKN detected an error when reading an interrupt coreload, or a coreload queued to the basic level.</td>
</tr>
<tr>
<td>SERR</td>
<td>INTERRUPT SAVE XXXX</td>
<td>BULKN detected an error writing VCORE to the interrupt save area (INSV).</td>
</tr>
<tr>
<td>CABR</td>
<td>INTERRUPT RESTORE XXXX</td>
<td>BULKN detected an error reading the interrupt save area into VCORE.</td>
</tr>
<tr>
<td>CRMON</td>
<td>BP MONITOR READ XXXX</td>
<td>BULKN detected an error reading the batch-process monitor bootstrap loader into VCORE.</td>
</tr>
<tr>
<td>BDRML</td>
<td>SPECIAL SAVE XXXX</td>
<td>BULKN detected an error writing VCORE to the special save area.</td>
</tr>
</tbody>
</table>

This setting causes a CALL EXIT to be executed.
ERROR MESSAGES

If the principal 1053 and all its backup units are offline, each attempt by the CE Coreload to type a message will cause a WAIT with a unique display in the A-register. These waits and their associated messages are as follows:

<table>
<thead>
<tr>
<th>Message</th>
<th>A-register</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Type out of CE switch settings)</td>
<td>/F001 (See Note)</td>
</tr>
<tr>
<td>CE CORELOAD</td>
<td>/F002</td>
</tr>
<tr>
<td>SET FUNC IN CE SWITCHES</td>
<td>/F003</td>
</tr>
<tr>
<td>DEVC OR UNIT NOT ON SYST</td>
<td>/F004</td>
</tr>
<tr>
<td>INVALID DEVICE CODE</td>
<td>/F005</td>
</tr>
<tr>
<td>INVALID DEVICE FOR SWITCH</td>
<td>/F006</td>
</tr>
<tr>
<td>NO DEVICE SELECTED</td>
<td>/F007</td>
</tr>
<tr>
<td>TURN ALL SWITCHES OFF TO EXIT</td>
<td>/F008</td>
</tr>
<tr>
<td>LOGICAL UNIT NO. TOO LARGE</td>
<td>/F009</td>
</tr>
<tr>
<td>INVALID DEVICE FOR SWITCH</td>
<td>/F010</td>
</tr>
<tr>
<td>CHANGE TO CONTINUE</td>
<td>/FFFF</td>
</tr>
<tr>
<td>(Status line for device unit)</td>
<td>/0001</td>
</tr>
<tr>
<td>SYST PTR ERR-DUMP ABORTED</td>
<td>/8001</td>
</tr>
</tbody>
</table>

Note: /F001 is displayed when a value is to be set in the CE switches (following SET FUNC IN CE SWITCHES).

The Q-register displays the most recent setting of the CE sense switches in bits 8 - 15. Bit 0 of the Q-register is set to 1 if the requested function was successfully completed.

CECLX BRANCH FUNCTIONS

The following functions enable you to branch to the CECLX communications adapter, 2311, or 2790 extensions. Each branch function can be executed by either the basic part of the CECLX program or the other extension.

Branch to Communications Adapter Extension

1. Set CE sense switches to 0000 0001.
2. Press console START.
This page intentionally left blank.
Branch to 2311 Extension

1. Set CE sense switches to 0000 0010.
2. Press console START.

Branch to 2790 Extension

1. Set CE sense switches to 00000011.
2. Press console START.

CECLX COMMUNICATIONS ADAPTER EXTENSION FUNCTIONS

The communications adapter extension of CECLX is entered through the basic program or one of the extensions of CECLX using the Branch to Communications Adapter Extension function described above. This extension enables you to do the following:

Dump Error Statistics Table

1. Set CE sense switches to:
   - 0010 ZYYY to dump without reset
   - 0011 ZYYY to dump and reset
   where:
   - Z = 0 and YYY = logical device number for the table, or
   - Z = 1 to dump and/or reset the Error Statistics Tables for all communications adapter units (YYY is not used).
2. Press console START.
   The dump is printed on the system printer.

Dump CE Error Log

1. Set CE sense switches to:
   - 0100 0000 to dump without reset
   - 0101 0000 to dump and reset
2. Press console START.
   To terminate the dump and cancel the reset functions, change any of the switch settings during the dump. The dump is printed on the system printer.

Set Up to Start/Stop CA Line Trace

1. Set CE sense switches to:
   - 0110 OYYY to set up to start trace, where YYY is the logical unit number of the CA line
   - 0111 OYYY to set up to stop trace, where YYY is not used
2. Press console START.

Dump Communications Adapter Trace Buffer

1. Set CE sense switches to 1000 0000.
2. Press console START.
   The dump is printed on the system printer.

This function should not be selected unless you have first initialized the trace buffer by the CECLX function "Start CA Line Trace." Note that the contents of the trace buffer will be destroyed by the use of the CE diagnostics or your own use of CE Core.
Return to Basic Part of CECLX

1. Set CE sense switches to 1110 0000.

2. Press console START.

Error Messages

<table>
<thead>
<tr>
<th>Message</th>
<th>A-Register</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT CA FUNCTION</td>
<td>/5004</td>
</tr>
<tr>
<td>INVALID FUNCTION FOR CA</td>
<td>/5005</td>
</tr>
<tr>
<td>DEVICE OR UNIT NOT ON SYSTEM</td>
<td>/5006</td>
</tr>
<tr>
<td>ERR STATISTICS TBL NOT DEFINED</td>
<td>/5007</td>
</tr>
<tr>
<td>SYS PTR ERR — DUMP ABORTED</td>
<td>/8001</td>
</tr>
<tr>
<td>(When a value is to be set in the CE sense switches)</td>
<td>/5001</td>
</tr>
</tbody>
</table>

CECLX 2311 EXTENSION FUNCTIONS

The 2311 extension of CECLX is entered through the basic program or one of the extensions using the "Branch to 2311 Extension" function. This extension enables you to do the following functions:

Dump Error Statistics Table

1. Set CE sense switches to:
   - 0010 ZYYY to dump without reset
   - 0011 ZYYY to dump and reset

   where:
   - \(Z = 0\) and \(YYY\) = logical device number for the table, or
   - \(Z = 1\) and \(YYY\) to dump and/or reset the Error Statistics Tables for all units of the type chosen (\(YYY\) is not used).

2. Press console START.

The dump is printed on the system printer.

Dump CE Error Log

1. Set CE sense switches to:
   - 0100 0000 to dump without reset
   - 0101 0000 to dump and reset

2. Press START.

To terminate the dump and cancel the reset function, change any of the switch settings during the dump. The dump is printed on the system printer.

Return to Basic Part of CECLX

1. Set CE sense switches to 1110 0000.

2. Press console START.
**Error Messages**

<table>
<thead>
<tr>
<th>Message</th>
<th>A-Register</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT 2311 FUNCTION</td>
<td>/6004</td>
</tr>
<tr>
<td>INVALID FUNCTION FOR 2311</td>
<td>/6005</td>
</tr>
<tr>
<td>DEVICE OR UNIT NOT ON SYSTEM</td>
<td>/6006</td>
</tr>
<tr>
<td>ERR STATISTICS TBL NOT DEFINED</td>
<td>/6007</td>
</tr>
<tr>
<td>SYS PTR ERR — DUMP ABORTED</td>
<td>/8001</td>
</tr>
<tr>
<td>(When a value is to be set in the sense switches)</td>
<td>/6001</td>
</tr>
</tbody>
</table>

**CECLX 2790 EXTENSION FUNCTIONS**

The 2790 extension of CECLX is entered through the basic program or one of the extensions by the "Branch to 2790 Extension" function. This enables you to do the following operations:

**Dump 2790 Error Counts**

1. Set the CE sense switches to:
   - 0010000x to dump without reset
   - 0011000x to dump and reset area station error counts

   where:
   - x = 0 for loop number 1
   - x = 1 for loop number 2

2. Press Console START

   The following message is printed:
   
   ENTER AREA ATN ADDR
   
3. Set the CE sense switches to the area station address in hexadecimal (80-FF) or to 0, which indicates that the error counts for all area stations are to be printed. The system error count for that loop will also be printed if the sense switch setting was 0 or /FF (system error counts not reset).

4. Press console START. The error counts are printed on the system printer.

   A minus sign printed before the count means that a threshold error has occurred during this terminal time out period.

   NOT ACTIVE printed instead of the loop system error count means that the 2790 coreload is not in main storage or the loop is not active.

**Dump 2790 Error Log (can be used as an error sort)**

1. Set the CE sense switches to:
   - 01000000 for loop 1
   - 01000001 for loop 2

2. Press console START

   The following message is printed:
   
   ENTER AREA STN ADDR
3. Set the CE sense switches to the area station address in hexadecimal (80-FF) or to 0, which indicates that the error log entries for all area stations are to be printed.

4. Press console START. The error log entries are printed on the system printer.

2790 Error Sort Program

In addition to the CE Coreload dump of the 2790 Error Log, which can be used to sort errors by area station number, an Error Sort Program is available for this purpose.

The Error Sort Program is part of the 2790 On Line Diagnostic Programs which the system requires to be stored on drive 0 of your system. They are provided to you by your customer engineer with control cards that store them as a batch-processing coreload. You must assure that this program is maintained on your system residence disk.

After the AREA STATION THRESHOLD EAC message (see "Recovery From Errors" in the 1800/2790 MPX Data Communication System Programming manual, Order Number GC26-3732) has been printed, the errors in the error log must be sorted so that you may determine if a faulty data entry unit needs to be replaced or if an area station needs maintenance.

You can sort the errors by executing the Error Sort Program as a batch-processing job as follows:

```
1-10 11-20 21-30 31-40 41-50 51-60
1234567890 1234567890 1234567890 1234567890 1234567890
```

where L is the loop number (1 or 2) and AAA is the area station address (0-126).

Additional EDIT cards may be run to obtain error sorts on additional area stations if you want.

The output of the Error Sort Program is a listing of the last eight errors logged against the requested area station, printed in this format:

```
// JOB 07 JUL 70 09.489 HRS
// XEQ @CEX@ FX 07 JUL 70 09.490 HRS
DATE 188
TIME DATE AS/DA CC/DA CSW AS/DA CC/DA LOOP TR
09.397 188 /82CO /0E1F /0400 /82CO /3E00 /01 /00
09.397 188 /82CO /0E1F /0400 /82CO /3E00 /01 /00
09.396 188 /82CO /0E1F /0400 /82CO /3E00 /01 /00
09.348 188 /82CO /0E1F /0400 /82CO /3E00 /01 /00
09.348 188 /82CO /0E1F /0400 /82CO /3E00 /01 /00
09.347 188 /82CO /0E1F /0400 /82CO /3E00 /01 /00
09.347 188 /82CO /0E1F /0400 /82CO /3E00 /01 /00
09.342 188 /82CO /0E1F /0400 /82CO /3E00 /01 /00
END OF ERROR LOG
```

- 120.2 1800 MPX Operating Procedures
Refer to the 2790 Adapter section of the 1800 Functional Characteristics manual, Order Number GA26-5918, for a detailed description of the Device Address, Channel Control Word, and Channel Sense Word.

The error sort information can be used to determine what action is required. If the device address (DV) values for four or more errors indicate the same device, the action described in this table is recommended:

<table>
<thead>
<tr>
<th>DV Value (polling frame)</th>
<th>Device</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>No Device</td>
<td>No device selected, check DV value in error frame for possible action. Check for NO PAPER EAC message. Have personnel at machine location check for visible typewriter malfunctions.</td>
</tr>
<tr>
<td>40</td>
<td>Printer</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>Local Badge Reader</td>
<td>Have supervisory personnel verify operation of the suspect device. If the test proves the device to be malfunctioning, it should be taken out of service and repair service notified.</td>
</tr>
<tr>
<td>81</td>
<td>1st Remote Badge Reader</td>
<td></td>
</tr>
<tr>
<td>82</td>
<td>2nd Remote Badge Reader</td>
<td></td>
</tr>
<tr>
<td>83</td>
<td>3rd Remote Badge Reader</td>
<td></td>
</tr>
<tr>
<td>84</td>
<td>Card Reader</td>
<td></td>
</tr>
<tr>
<td>88</td>
<td>Keyboard</td>
<td></td>
</tr>
<tr>
<td>8C</td>
<td>Digital Device (OEM)</td>
<td></td>
</tr>
<tr>
<td>CO-DF</td>
<td>DEU</td>
<td>Decode the specific data entry unit (DEU) address from the DV value and have that unit tested or replaced according to established DEU service procedures.</td>
</tr>
</tbody>
</table>

Reset the 2790 Error Log

1. Set the CE sense switches to:
   
   01010000

2. Press console START.

   The following message is printed:

   2790 ERROR LOG RESET REQUESTED. IF OK, TURN ON CE SENSE SWITCH 15 AND PRESS START

3. Set CE sense switch 15 ON. Do not change the other sense switches.

4. Press console START

   Zeros will be written on each sector of the 2790 error log file (E2790).

Start/Reset a 2790 Segment

1. Set the CE sense switches to:

   011ABCD0 to start segments on loop 1
   011ABCD1 to start segments on loop 2
where:

A, B, C, or D is 0 to start the respective segment or 1 if the segment is not to be started.

2. Press console START.

The following message is printed on the system printer:

2790 SEGMENT START COMPLETION CODE = /00XX

If the completion code is /0001, the start was performed successfully. Note that this function will also restore a bypassed area station. A completion code of /0002-/001F indicates an error condition. A description of these completion codes can be found under "Completion Code Parameter for LACCN" in the 1800/2790 MPX Data Communication System Programming manual, Order Number GC26-3732.

Certain error conditions cause an Area Station to go into Send Address Mode. If you attempt to restart a segment where this condition exists the address of the Area Station in Send Address Mode is reported with the following message:

AREA STN IN SEND ADDR NODE IS /XX

Set On/Offline Status

1. Set the CE sense switches to:

1000000Y to take loop off line
1000001Y to put the loop on line

where:

Y = 0 for loop 1
Y = 1 for loop 2

2. Press console START.

The following message is printed on the system printer:

YOU REQUESTED LOOP X ON/OFF. IF OK TURN SW 11 AND PUSH START.

If switch 11 is turned on the following message is printed:

COMPLETION CODE /00XX

If the completion code is /0001, the function was performed successfully. A completion code of /0002 - /001F indicates an error condition. See the 1800/2790 MPX Data Communication System Programming manual, Order Number GC26-3732, for a description of these completion codes.

Return to Basic Part of CECLX

1. Set the CE sense switches to 1110 0000.

2. Press console START.

ERROR MESSAGES

If the 1053 (that is defined to be either the system and/or list printer) and all of its backup units are offline, each attempt by CECLX to type a message will cause a WAIT with a unique display in the A-register. These waits and their associated messages are as follows:
<table>
<thead>
<tr>
<th>Message</th>
<th>A-Register</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT 2790 FUNCTION</td>
<td>/7001 (See Note)</td>
</tr>
<tr>
<td>INVALID FUNCTION FOR 2790</td>
<td>/7004</td>
</tr>
<tr>
<td>ENTER AREA STN ADDR</td>
<td>/7005</td>
</tr>
<tr>
<td>2790 ERR LOG FILE NOT DEFINED</td>
<td>/7006</td>
</tr>
<tr>
<td>DISK ERROR</td>
<td>/7007</td>
</tr>
<tr>
<td>2790 ERROR LOG RESET REQUIRED. IF OK, TURN ON CE SENSE SWITCH 15 AND PRESS START.</td>
<td>/7008</td>
</tr>
<tr>
<td>..RESET COMPLETED</td>
<td>/7009</td>
</tr>
<tr>
<td>SYS PTR ERR -DUMP ABORTED</td>
<td>/700A</td>
</tr>
</tbody>
</table>

Note: /7001 is displayed when a value is to be set in the CE sense switches.
EXAMPLES OF CE CORELOAD OPERATION

Figure 15 is an example of how to build and use CECLD. In the example, CECLD is built into a batch-processing coreload and used to take a 1053 printer off line and assign another printer to cover for it.

```plaintext
// JOB   00 JAN 00 00.079 HRS
// XEQ CECLD L   00 JAN 00 00.080 HRS  Build CECLD

*CCEND
MPX, BUILD CECLD
CORE LOAD MAP
TYPE NAME ARG1 ARG2
*CDW TABLE 8002  0012
*F10 TABLE 8014  001E
*CNT TABLE 8032  0004
MAIN CECLD 8036
CLNT CECLD 8034
CORE 83F2  7COE
MPX, CECLD LD XQ

CE CORELOAD

SET FUNC IN CE SWITCHES 00100001 Select device type — 1816/1053 printer
1 TYPE01 ON 0000
2 TYPE02 ON 0000
3 TYPE03 ON 0000
4 TYPE04 ON 0000
5 TYPE05 ON 0000

SET FUNC IN CE SWITCHES 01000001 Print out status of 1816/1053s
2 TYPE02 OFF 0000

SET FUNC IN CE SWITCHES 11010001 Take 1053/2 off line
2 TYPE03 ON 0000

SET FUNC IN CE SWITCHES 00100001 Assign logical unit number 2 to 1053/3
1 TYPE01 ON 0000
2 TYPE02 ON 0000
3 TYPE03 ON 0000
4 TYPE04 ON 0000
5 TYPE05 ON 0000

SET FUNC IN CE SWITCHES 00000000 Print out status of 1816/1053s

Figure 15. Using CECLD
```
Figure 16 is an example of how to build and use CECLX. In the example, CECLX is built into a batch-processing coreload and used to print the Error Statistics Tables and CE Error Logs for two communications adapters, two 2311 drives, and a 2790 data communication system with two loops.

```
// JOB 00 JAN 00 00.204 HRS
// EQQ CECLX L 00 JAN 00 00.204 HRS
*CCEND
MPX, BUILD CECLX
CORE LOAD MAP
TYPE NAME ARG1 ARG2
*CM TABLE 8002 0012
*FIO TABLE 8014 001E
*CNT TABLE 8032 0004
MAIN CECLX 8036
CLNT CECLX 8034
CALL EBPST 8072
CALL EPRST 8072
CALL PRST REC
CORE BE68 7198
MPX, CECLX LD XQ
CE CORELOAD
SET FUNC IN CE SWITCHES 00000001
SELECT CA FUNCTION 00101000
ERROR STATISTICS TABLE FOR CA LINE NO. 0, DAY 000, HOUR 17
   HARDWARE - NONE
   MASTER - NONE
   SLAVE - NONE

ERROR STATISTICS TABLE FOR CA LINE NO. 1, DAY 000, HOUR 00
   HARDWARE - NONE
   MASTER
   CODE COUNT
   /0031 /0091
   /0035 /0042
   /0036 /0004
   /0037 /0010
   SLAVE
   CODE COUNT
   /0043 /0005
   /0046 /0004
   /0047 /0003

ERROR STATISTICS TABLE FOR CA LINE NO. 2, DAY 000, HOUR 65
   HARDWARE
   CODE COUNT
   /0005 /0007
   MASTER
   CODE COUNT
   /0031 /0001
   /0035 /0007
   /0037 /0002
   SLAVE - NONE

SELECT CA FUNCTION 01000000
COMMUNICATIONS ADAPTER CE ERROR LOG DAY 000, HOUR 00, MSGS LOGGED 006
TIME NO. OF OPERATING FUNCTION ERROR
DAY HOUR ERRORS DSW CODE CODE
.044 1b /0007 /2080 /2501 /0041
044 1b /0006 /10A0 /2501 /0043
044 1b /0001 /08A0 /2501 /0042
044 1b /0001 /10A0 /2501 /0043
044 1b /0001 /08A0 /2501 /0042
044 1b /0001 /10A0 /2501 /0043
```

Figure 16. Using CECLX
SELECT CA FUNCTION 10000000 ———— Print CA trace

| R 0000 | DSW 1000 FUNCTION CODE 1304 COMPLETION CODE 0001 |
| R 0000 | DSW 1040 FUNCTION CODE 2204 COMPLETION CODE 0001 |
| R 0000 | DSW 1001 20FF |
| R 0000 | DSW 2000 FUNCTION CODE 2504 COMPLETION CODE 0041 |
| R 0000 | DSW 1070 FF00 |
| R 0000 | DSW 1000 FUNCTION CODE 2504 COMPLETION CODE 0001 |
| R 0000 | DSW 1070 FF00 |
| R 0002 | DSW 0800 FUNCTION CODE 3704 COMPLETION CODE 0001 |

The column labeled 2311 ARM/HEAD contains the arm and head position at the time of the error. This is determined by reading the home address after the error.

The column labeled DESIRED CC/TT contains the cylinder and track position that was specified by the failing CCW.

SELECT 2311 FUNCTION 01000000 ———— Print 2311 EST

<table>
<thead>
<tr>
<th>CODE</th>
<th>COUNT</th>
</tr>
</thead>
<tbody>
<tr>
<td>/0001</td>
<td>/0000011B</td>
</tr>
<tr>
<td>/0002</td>
<td>/0003</td>
</tr>
<tr>
<td>/0003</td>
<td>/0003</td>
</tr>
</tbody>
</table>

Figure 16. Using CECLX (Cont.)

Lines of the trace dump labeled R are received data. A T indicates transmitted data. The first word is the indicator/byte count word. The remaining words contain the data transmitted over the line. Where the message length exceeds 14 characters, an ellipsis indicates characters omitted from the middle of the message. The function code shown in the trace dump is from the BSCIO call list, not the actual machine IOCC.

CE Coreload Programs (CECLD, CECLX) 122.1
Table A. Communications Adapter Completion and Error Codes

This table lists all completion and error codes used by the BSCIO subroutine and indicates where each is used or recorded.

<table>
<thead>
<tr>
<th>Completion code</th>
<th>Retry counter</th>
<th>Error Statistics Table</th>
<th>CE Error Log</th>
<th>EAC message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X X</td>
<td>X X X X X X</td>
<td>/0001</td>
<td></td>
<td>X /0005</td>
<td>Operation successful</td>
</tr>
<tr>
<td>X</td>
<td>X X X X X</td>
<td>/0002</td>
<td></td>
<td>X /0003</td>
<td>Device off line</td>
</tr>
<tr>
<td>X</td>
<td>X X X X X</td>
<td>/0003</td>
<td></td>
<td>X /0004</td>
<td>Modem not ready</td>
</tr>
<tr>
<td>X X X X X X</td>
<td>/0004</td>
<td></td>
<td></td>
<td>X /0005</td>
<td>Data parity</td>
</tr>
<tr>
<td>X X X X X X</td>
<td>/0006</td>
<td></td>
<td></td>
<td>X /0006</td>
<td>Data check</td>
</tr>
<tr>
<td>X X X X X X</td>
<td>/0007</td>
<td></td>
<td></td>
<td>X /0008</td>
<td>Data overrun</td>
</tr>
<tr>
<td>X X</td>
<td>X /0008</td>
<td></td>
<td></td>
<td>X /0009</td>
<td>No-response</td>
</tr>
<tr>
<td>X X</td>
<td>/0011</td>
<td></td>
<td></td>
<td>X /0012</td>
<td>List aborted-abort discontinue</td>
</tr>
<tr>
<td>X</td>
<td>/0013</td>
<td></td>
<td></td>
<td>X /0014</td>
<td>Initial not done</td>
</tr>
<tr>
<td>X</td>
<td>/0015</td>
<td></td>
<td></td>
<td>X /0016</td>
<td>Wrong state of line</td>
</tr>
<tr>
<td>X</td>
<td>/0017</td>
<td></td>
<td></td>
<td>X /0018</td>
<td>ID function not done</td>
</tr>
<tr>
<td>X X</td>
<td>/0019</td>
<td></td>
<td></td>
<td>X /0019</td>
<td>Two RVIs in a row</td>
</tr>
<tr>
<td>X X</td>
<td>/0020</td>
<td></td>
<td></td>
<td>X /0021</td>
<td>List aborted by restart</td>
</tr>
<tr>
<td>X X</td>
<td>/0022</td>
<td></td>
<td></td>
<td>X /0023</td>
<td>RVI received</td>
</tr>
<tr>
<td>X X</td>
<td>/0024</td>
<td></td>
<td></td>
<td>X /0025</td>
<td>DISC received</td>
</tr>
<tr>
<td>X</td>
<td>/0026</td>
<td></td>
<td></td>
<td>X /0027</td>
<td>EOT received</td>
</tr>
<tr>
<td>X X X X X X</td>
<td>/0028</td>
<td></td>
<td></td>
<td>X /0029</td>
<td>Poll address received</td>
</tr>
<tr>
<td>X X</td>
<td>/0031</td>
<td></td>
<td></td>
<td>X /0032</td>
<td>Selection address received</td>
</tr>
<tr>
<td>X X X X X X</td>
<td>/0033</td>
<td></td>
<td></td>
<td>X /0034</td>
<td>Timeout as master</td>
</tr>
<tr>
<td>X X X X X X</td>
<td>/0035</td>
<td></td>
<td></td>
<td>X /0036</td>
<td>Invalid sequence received</td>
</tr>
<tr>
<td>X X X X X X</td>
<td>/0037</td>
<td></td>
<td></td>
<td>X /0038</td>
<td>Message resulted in negative response</td>
</tr>
<tr>
<td>X X X X X X</td>
<td>/0039</td>
<td></td>
<td></td>
<td>X /0040</td>
<td>Forward abort by slave</td>
</tr>
<tr>
<td>X X X X X X</td>
<td>/0041</td>
<td></td>
<td></td>
<td>X /0042</td>
<td>WACK received</td>
</tr>
<tr>
<td>X X X X X X</td>
<td>/0043</td>
<td></td>
<td></td>
<td>X /0044</td>
<td>TTD sent</td>
</tr>
<tr>
<td>X X X X X X</td>
<td>/0045</td>
<td></td>
<td></td>
<td>X /0046</td>
<td>Message transmitted successfully</td>
</tr>
<tr>
<td>X X X X X X</td>
<td>/0047</td>
<td></td>
<td></td>
<td>X /0048</td>
<td>Message aborted - (Data..ENQ)</td>
</tr>
<tr>
<td>X X</td>
<td>/0050</td>
<td></td>
<td></td>
<td>X /0051</td>
<td>Storage protect</td>
</tr>
<tr>
<td>X</td>
<td>/0052</td>
<td></td>
<td></td>
<td>X /0053</td>
<td>Command reject</td>
</tr>
<tr>
<td>X</td>
<td>/0054</td>
<td></td>
<td></td>
<td>X /0055</td>
<td>Call error</td>
</tr>
</tbody>
</table>

122. 2 1800 MPX Operating Procedures
<table>
<thead>
<tr>
<th>Condition</th>
<th>Retries</th>
<th>Error Code</th>
<th>CE Error Log</th>
<th>ERP Statistics Table Word</th>
<th>EAC Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface Control Check</td>
<td>1</td>
<td>RS/RL</td>
<td>Yes</td>
<td>2</td>
<td>Yes</td>
</tr>
<tr>
<td>Channel Data Check</td>
<td>1</td>
<td>RS/RL</td>
<td>Yes</td>
<td>2</td>
<td>Yes</td>
</tr>
<tr>
<td>Not Operational</td>
<td>1</td>
<td>RS/RL</td>
<td>Yes</td>
<td>2</td>
<td>Yes</td>
</tr>
<tr>
<td>Equipment Check</td>
<td>0</td>
<td>RS/RL</td>
<td>No</td>
<td>3</td>
<td>Yes</td>
</tr>
<tr>
<td>No Record Found</td>
<td>n</td>
<td>9</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Seek Check</td>
<td>n</td>
<td>8</td>
<td>Yes</td>
<td>8</td>
<td>Yes</td>
</tr>
<tr>
<td>Intervention Required</td>
<td>0</td>
<td>3</td>
<td>No</td>
<td>8</td>
<td>Yes</td>
</tr>
<tr>
<td>Bus Out Check</td>
<td>n</td>
<td>4</td>
<td>Yes</td>
<td>4</td>
<td>Yes</td>
</tr>
<tr>
<td>Data Check</td>
<td>n</td>
<td>6</td>
<td>Yes</td>
<td>6</td>
<td>Yes</td>
</tr>
<tr>
<td>Overrun</td>
<td>n</td>
<td>7</td>
<td>Yes</td>
<td>7</td>
<td>Yes</td>
</tr>
<tr>
<td>Missing Address Marker</td>
<td>n</td>
<td>RS/RL</td>
<td>Yes</td>
<td>9</td>
<td>Yes</td>
</tr>
<tr>
<td>Command Reject</td>
<td>0</td>
<td>RS/RL</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Track Overrun</td>
<td>0</td>
<td>5</td>
<td>No</td>
<td>5</td>
<td>Yes</td>
</tr>
<tr>
<td>End of Cylinder</td>
<td>0</td>
<td>A</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>File Protect</td>
<td>0</td>
<td>D</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Overflow Incomplete</td>
<td>0</td>
<td>RS/RL</td>
<td>No</td>
<td>A</td>
<td>Yes</td>
</tr>
<tr>
<td>Program Check</td>
<td>n</td>
<td>RS/RL</td>
<td>Yes</td>
<td>D</td>
<td>Yes</td>
</tr>
<tr>
<td>Unit Exception</td>
<td>0</td>
<td>C</td>
<td>No</td>
<td>C</td>
<td>No</td>
</tr>
<tr>
<td>Incorrect Length</td>
<td>0</td>
<td>B</td>
<td>No</td>
<td>B</td>
<td>No</td>
</tr>
<tr>
<td>COMMAND</td>
<td>COMMAND CODE</td>
<td>DATA ADDRESS</td>
<td>COUNT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------------</td>
<td>--------------</td>
<td>--------------</td>
<td>-------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td>83 03</td>
<td>0000-0011</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seek</td>
<td>07 07</td>
<td>0000-0011</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seek Cylinder</td>
<td>11 0b</td>
<td>0000-1011</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seek Head</td>
<td>11 0b</td>
<td>0000-1011</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Set-Fr-Mask</td>
<td>31 17</td>
<td>0001-1111</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Space Count</td>
<td>02 02</td>
<td>0001-0011</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transfer in Cylinder</td>
<td>68 08</td>
<td>XXX-XXXX1000</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Restart</td>
<td>58 13</td>
<td>0001-0011</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Restore</td>
<td>23 17</td>
<td>0001-0011</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sense</td>
<td>04 04</td>
<td>0000-0100</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Switching</td>
<td>140 14</td>
<td>1001-0190</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reserve Device</td>
<td>150 04</td>
<td>1011-0190</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Search</td>
<td>51 29</td>
<td>0011-1001</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identify Equals</td>
<td>40 31</td>
<td>0011-0001</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identify High</td>
<td>61 51</td>
<td>0110-0001</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identify High Or High</td>
<td>113 71</td>
<td>0111-0001</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Key Equ</td>
<td>41 29</td>
<td>0010-1001</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Key High</td>
<td>17 49</td>
<td>0100-1001</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Key Equal Or High</td>
<td>106 59</td>
<td>0110-1001</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Key and Data Equal</td>
<td>46 27</td>
<td>0010-1101</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Key and Data Equal Or High</td>
<td>106 60</td>
<td>0110-1101</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Continue Scan High</td>
<td>27 25</td>
<td>0010-0001</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Continue Scan High Or High</td>
<td>85 45</td>
<td>0101-0001</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Continue Scan No Compare</td>
<td>85 55</td>
<td>0101-0001</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Continue Scan No Compare Or High</td>
<td>171 65</td>
<td>0111-0001</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Read Address</td>
<td>01 1A</td>
<td>0001-1010</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Count</td>
<td>18 12</td>
<td>0001-0010</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Record RD</td>
<td>22 16</td>
<td>0010-0110</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data</td>
<td>04 04</td>
<td>0000-0110</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Key and Data</td>
<td>14 0E</td>
<td>0000-1110</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Count, Key and Data</td>
<td>28 12</td>
<td>0011-1110</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Initial Program End (FPI)</td>
<td>13 02</td>
<td>0000-0010</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Write</td>
<td>25 19</td>
<td>0010-0001</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Record RD</td>
<td>21 15</td>
<td>0001-0101</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Count, Key and Data</td>
<td>29 12</td>
<td>0001-1101</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>General Count, Key and Data</td>
<td>31 01</td>
<td>0000-0010</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data</td>
<td>69 29</td>
<td>0000-0101</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Key and Data</td>
<td>13 30</td>
<td>0010-0011</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Erase</td>
<td>17 11</td>
<td>0001-0011</td>
<td>X</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* Special Features
X = Not Significant
Table D. 2841/2311 Status and Sense Bytes

<table>
<thead>
<tr>
<th>CSW Status Byte</th>
<th>Sense Byte 0</th>
<th>Sense Byte 1</th>
<th>Sense Byte 2</th>
<th>Sense Byte 3</th>
<th>Sense Byte 4</th>
<th>Sense Byte 5*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit 0 Attention</td>
<td>Command</td>
<td>Data Check</td>
<td>Unsafe</td>
<td>Ready</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Reject</td>
<td>in Count Area</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit 1 Status Modifier</td>
<td>Intervention</td>
<td>Track Overrun</td>
<td>not used</td>
<td>On Line</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Required</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit 2 Control Unit End</td>
<td>Bus-Out Parity</td>
<td>End of Cylinder</td>
<td>Serializer Check</td>
<td>Unsafe</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Parity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit 3 Busy</td>
<td>Equipment</td>
<td>Invalid</td>
<td>not used</td>
<td>not used</td>
<td>0</td>
<td>See Note</td>
</tr>
<tr>
<td></td>
<td>Check</td>
<td>Sequence</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit 4 Channel End</td>
<td>Data Check</td>
<td>No Record</td>
<td>ALU Check</td>
<td>On Line</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Found</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit 5 Device End</td>
<td>Overrun</td>
<td>File</td>
<td>Unselected</td>
<td>End of Cylinder</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Protected</td>
<td>File Status</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit 6 Unit Check</td>
<td>Track Condition Check</td>
<td>Missing Address Marker</td>
<td>not used</td>
<td>not used</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Seek Check</td>
<td>Overflow</td>
<td>not used</td>
<td>Seek</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Incomplete</td>
<td></td>
<td>Incomplete</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Sense Byte 5 is used only for the overflow feature. The setting of the bits is determined by the conditions existing at the time of the interruption and the type of operation in progress. For further information on the bit configurations, refer to the description in the text of the manual.
SELECT 2311 FUNCTION 00000011
SELECT 2790 FUNCTION 00110000
ENTER AREA STN ADDR 00000000

2790 ERROR COUNT LOOP NO. 1

<table>
<thead>
<tr>
<th>AREA STN</th>
<th>COUNT</th>
</tr>
</thead>
<tbody>
<tr>
<td>/80</td>
<td>0600</td>
</tr>
<tr>
<td>/81</td>
<td>0601</td>
</tr>
<tr>
<td>/82</td>
<td>0614</td>
</tr>
<tr>
<td>/83</td>
<td>0603</td>
</tr>
<tr>
<td>/84</td>
<td>0605</td>
</tr>
<tr>
<td>/85</td>
<td>0601</td>
</tr>
<tr>
<td>/86</td>
<td>0600</td>
</tr>
</tbody>
</table>

END--COUNT PRINTED ONLY IF DIFFERENT FROM PRECEDING.

SELECT 2790 FUNCTION 01000000
ENTER AREA STN ADDR 00000000

2790 ERROR LOG AREA STN=ALL DAY 155

<table>
<thead>
<tr>
<th>DATE</th>
<th>TIME</th>
<th>AS/DA</th>
<th>CC/DA</th>
<th>CSW</th>
<th>AS/DA</th>
<th>CC/DA</th>
<th>LOOP</th>
<th>TRAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>155</td>
<td>08.213</td>
<td>/8140</td>
<td>/0000</td>
<td>/4000</td>
<td>/8140</td>
<td>/0040</td>
<td>/01</td>
<td>00</td>
</tr>
<tr>
<td>155</td>
<td>08.212</td>
<td>/8140</td>
<td>/0000</td>
<td>/4000</td>
<td>/8140</td>
<td>/0040</td>
<td>/01</td>
<td>00</td>
</tr>
<tr>
<td>155</td>
<td>08.212</td>
<td>/8140</td>
<td>/0000</td>
<td>/4000</td>
<td>/8140</td>
<td>/0040</td>
<td>/01</td>
<td>00</td>
</tr>
<tr>
<td>155</td>
<td>08.210</td>
<td>/8140</td>
<td>/0000</td>
<td>/4000</td>
<td>/8140</td>
<td>/00160</td>
<td>/01</td>
<td>00</td>
</tr>
</tbody>
</table>

END OF ERROR LOG

SELECT 2790 FUNCTION 01010000

2790 ERROR LOG RESET REQUESTED. IF OK, TURN ON CE SENSE SW 15 AND PRESS START. RESET COMPLETED.

SELECT 2790 FUNCTION 01100000

2790 SEGMENT START, COMPLETION CODE /0001

Figure 16. Using CECLX (Cont.)
1442 Reader Diagnostic

The MPX system contains a test to diagnose 1442 card read punch read errors and stacker select while the real-time system is operating. This program is part of the Supervisor, Phase SUPD. The 1442 must be taken off line by the CE Coreload before the test can be run.

When the 1442 is taken off line by the CE Coreload, the following message is printed on the system printer, and background processing is terminated:

// ERROR END OF ALL JOBS

OPERATING PROCEDURES

1. Clear the 1442 card reader.

2. Turn program switch 7 ON, CE switch 15 ON, and press CONSOLE INTERRUPT.

3. The following messages are printed:

   CUST ENG 1442 C002 CE SWS ALL OFF TO EXIT
   CUST ENG 1442 C001 LOAD TEST + MAKE READY
   CUST ENG 1442 E002 READER NOT READY

4. Load your punched reader test complement cards in the reader and press reader START. Do not put a blank card behind the test cards.

   The cards for the reader test must be punched so the first 40 columns complement the second 40 columns. For example, if there is a 12 punch in column 1, column 80 must contain an 11 through 9 punch. If column 2 is laced (all punches), column 79 must be blank. A complement bit pattern must be used for all complementary pairs of columns on the card. Figure 17 illustrates a typical reader test complement card.

   ![Figure 17. Sample Complement Input Card for 1442 Reader Diagnostic](image)

   Figure 17. Sample Complement Input Card for 1442 Reader Diagnostic

1800 MPX Operating Procedures
When the complement cards are placed in the reader and the reader is made ready, the reader diagnostic issues a call to bring the reader on line, read a card, and put the reader back off line. The card that was read is first checked for picking up a digit in column 40 or 41 and then checked for dropping a digit in these same two columns. Columns 1 and 80 are the last columns checked. Any time an error is detected, it is typed as an E000 message (see "1442 Diagnostic Error Messages").

To suppress printing of all messages, turn CE switch 8 ON.

5. Every other card that is read is stacker selected. Following the reading of the complement cards, the program will again print the reader not ready message. To read the last card, press reader START. The following message is printed:

CUST ENG 1442 C003 READER LAST CARD

6. To return the reader to online status, turn CE switch 15 OFF and call the CE Coreload, or if the CE Coreload was loaded from cards, do a cold start.

1442 DIAGNOSTIC ERROR MESSAGES

CUST ENG 1442 E000 DIGIT XX PICKED COL YY OR YY

or

CUST ENG 1442 E003 DIGIT XX DROPPED COL YY OR YY

(XX is the card digit that was picked or dropped during reading. YY is the column in which the error was found.)

CUST ENG 1442 E001 READER OFF-LINE

(Reader was placed online by diagnostic but was found offline by read.)

CUST ENG 1442 E002 READER NOT READY

(The reader became not ready following the reading of the last card.)

CUST ENG 1442 E003 READER PARITY ERR

(A parity error occurred during the last read command.)

CUST ENG 1442 E004 READER FEED CHECK

(A feed check occurred during the last read command.)

CUST ENG 1442 E005 READER READ CHECK

(A read register check occurred during the reading of the last card.

CUST ENG 1442 E006 INVALID CONDITION

(CARDZ returned on invalid error parameter.)
This page intentionally left blank.
Appendix A. System Loader Assignment Cards

The assignment card is used to specify the I/O device and machine function assignments to interrupt levels and bit position on the level. The assignments are in the form of Interrupt Assignment Codes (IAC), which are fixed for each device, and Logical Unit Numbers (LUN), which are selected by you for linkage for the FORTRAN programs. You must punch the assignment cards.

An Interrupt Assignment Code (IAC) has been assigned by IBM for all possible devices. These assignments cannot be changed. Assignment cards for interrupt levels greater than 23 should not be made, except for the special use of 99. If a value greater than 23 and other than 99 is punched in the assignment card, an error message is printed and the assignment is ignored.

The possible assignment values for the FORTRAN Logical Unit Numbers (LUN) are 01 through 44. The same LUN cannot be assigned to more than one device nor can a device have more than one LUN assigned to it.

Table 1 lists IAC/LUN assignments for the MPX system. The format of the assignment card is shown in Table 2.

Notes:
1. Continuation of bit assignments for a level onto another card is done as follows:
   - Place any nonblank character in column 72.
   - Place the total number of IAC codes assigned for the level in column 4 of all cards for the level, for example:
   - cc 1 4 7 72
   - 99 04 24, 25/20, 27 X
   - 99 04 44

2. DMON and all its phases must be the first module loaded in a //TOTAL SYSTEM LOAD operation. If an *LDDSK card and an *ASSIGNMENT card are both being used in the same load or update operation, the *LDDSK must follow the *ASSIGNMENT.
<table>
<thead>
<tr>
<th>Device</th>
<th>IAC</th>
<th>Possible LUN Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interval timers</td>
<td>00</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>First 1816/1053 on printer group 1</td>
<td>01</td>
<td>01 through 44</td>
</tr>
<tr>
<td>First 1442 card read punch</td>
<td>02</td>
<td>01 through 44</td>
</tr>
<tr>
<td>1054/1055 paper tape units</td>
<td>03</td>
<td>01 through 44</td>
</tr>
<tr>
<td>First 1810 disk or first mapped 1810**</td>
<td>04</td>
<td>01 through 44</td>
</tr>
<tr>
<td>1627 plotter</td>
<td>05</td>
<td>01 through 44</td>
</tr>
<tr>
<td>1443 printer</td>
<td>06</td>
<td>01 through 44</td>
</tr>
<tr>
<td>First 2790 adapter</td>
<td>07</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Second 1810 disk or second mapped 1810**</td>
<td>08</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Third 1810 disk or third mapped 1810**</td>
<td>09</td>
<td>01 through 44</td>
</tr>
<tr>
<td>First analog-to-digital converter (ADC)</td>
<td>10</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Digital input</td>
<td>11</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Digital analog output</td>
<td>12</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>System/360 channel adapter</td>
<td>13</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>First 1816/1053 on printer group 2</td>
<td>14</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Analog input expander, second analog-to-digital converter</td>
<td>15</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Second 1442 card read punch</td>
<td>16</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>2841 control unit</td>
<td>17</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Second 2790 adapter</td>
<td>18</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Fourth communications adapter line 0</td>
<td>19</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>First communications adapter line 0</td>
<td>20</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Second communications adapter line 0</td>
<td>21</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Third communications adapter line 0</td>
<td>22</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Fourth mapped 1810*</td>
<td>23</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Fifth mapped 1810*</td>
<td>24</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Sixth mapped 1810*</td>
<td>25</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Seventh mapped 1810*</td>
<td>26</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Eighth mapped 1810*</td>
<td>27</td>
<td>01 through 44</td>
</tr>
<tr>
<td>RPQ***</td>
<td>28</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Console interrupts</td>
<td>29</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Process interrupts</td>
<td>30</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>First comparator</td>
<td>31</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Second comparator</td>
<td>32</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Second 1053 on printer group 1</td>
<td>33</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Third 1053 on printer group 1</td>
<td>34</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Fourth 1053 on printer group 1</td>
<td>35</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Second 1053 on printer group 2</td>
<td>36</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Third 1053 on printer group 2</td>
<td>37</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Fourth 1053 on printer group 2</td>
<td>38</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Third 1053 on printer group 3</td>
<td>39</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Fourth 1053 on printer group 4</td>
<td>40</td>
<td>01 through 44</td>
</tr>
<tr>
<td>1816 keyboard on printer group 1*</td>
<td>41</td>
<td>01 through 44</td>
</tr>
<tr>
<td>1816 keyboard on printer group 2*</td>
<td>42</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Magnetic tape drive 2*</td>
<td>43</td>
<td>01 through 44</td>
</tr>
<tr>
<td>First communications adapter line 1</td>
<td>44</td>
<td>01 through 44</td>
</tr>
<tr>
<td>Second communications adapter line 1</td>
<td>45</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Third communications adapter line 1</td>
<td>46</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>Fourth communications adapter line 1</td>
<td>47</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td>RPQX***</td>
<td>48</td>
<td>No LUN assignable</td>
</tr>
<tr>
<td></td>
<td>49-63</td>
<td>No LUN assignable</td>
</tr>
</tbody>
</table>

*indicates that a dummy interrupt level number 99 is required.
**indicates that a dummy interrupt level number 99 is required if a mapped 1810 is assigned.
***RPQX is an extension of RPQ.
Table 2. Assignment Card Format

<table>
<thead>
<tr>
<th>Card Column</th>
<th>Possible Values</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>00 through 23, or 99</td>
<td>The entry in this field specifies the interrupt level to which this card is applicable. A separate card must be made for each interrupt level used. An entry of 99 in this field is a dummy interrupt level entry to provide FORTRAN linkages for the 1816 keyboards, the second magnetic tape unit, or mapped 1810 drives.</td>
</tr>
<tr>
<td>3</td>
<td>Blank</td>
<td>Must be blank</td>
</tr>
<tr>
<td>4-5</td>
<td>01 through 16</td>
<td>The entry in this field specifies the number of Interrupt Level Status Word (ILSW) bits that are assigned to this interrupt level.</td>
</tr>
<tr>
<td>6</td>
<td>Blank</td>
<td>Must be blank</td>
</tr>
<tr>
<td>7-72</td>
<td>A,B/X,C,...</td>
<td>A,B/X,C,... represent a group of IAC and IAC/LUN assignments. See Table 1 for possible values. The IAC and IAC/LUN assignments must be separated by commas. An element of the group must be either an IAC only or an IAC/LUN combination. An IAC/LUN combination must consist of two numbers separated by a slash. Some IACs have no LUN assignable to them (see Table 2). If a LUN is assignable to an IAC, but no LUN is assigned, the System Loader equates LUN to the IAC by default. A particular LUN cannot be assigned to more than one device. If column 72 contains a nonblank character, a continuation card must be included.</td>
</tr>
</tbody>
</table>

Table 1 for possible values. The IAC and IAC/LUN assignments must be separated by commas. An element of the group must be either an IAC only or an IAC/LUN combination. An IAC/LUN combination must consist of two numbers separated by a slash. Some IACs have no LUN assignable to them (see Table 2). If a LUN is assignable to an IAC, but no LUN is assigned, the System Loader equates LUN to the IAC by default. A particular LUN cannot be assigned to more than one device. If column 72 contains a nonblank character, a continuation card must be included.
<table>
<thead>
<tr>
<th>SECTOR ADDRESS</th>
<th>SECTOR ADDRESS</th>
<th>CYLINDER ADDRESS</th>
<th>CYLINDER ADDRESS</th>
<th>SECTOR ADDRESS</th>
<th>CYLINDER ADDRESS</th>
<th>SECTOR ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000</td>
<td>0000</td>
<td>0000</td>
<td>0000</td>
<td>00000</td>
<td>0000</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0001</td>
<td>0002</td>
<td>0003</td>
<td>00000</td>
<td>0004</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0005</td>
<td>0006</td>
<td>0007</td>
<td>00000</td>
<td>0008</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0009</td>
<td>0010</td>
<td>0011</td>
<td>00000</td>
<td>0012</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0013</td>
<td>0014</td>
<td>0015</td>
<td>00000</td>
<td>0016</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0017</td>
<td>0018</td>
<td>0019</td>
<td>00000</td>
<td>0020</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0021</td>
<td>0022</td>
<td>0023</td>
<td>00000</td>
<td>0024</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0025</td>
<td>0026</td>
<td>0027</td>
<td>00000</td>
<td>0028</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0029</td>
<td>0030</td>
<td>0031</td>
<td>00000</td>
<td>0032</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0033</td>
<td>0034</td>
<td>0035</td>
<td>00000</td>
<td>0036</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0037</td>
<td>0038</td>
<td>0039</td>
<td>00000</td>
<td>0040</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0041</td>
<td>0042</td>
<td>0043</td>
<td>00000</td>
<td>0044</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0045</td>
<td>0046</td>
<td>0047</td>
<td>00000</td>
<td>0048</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0049</td>
<td>0050</td>
<td>0051</td>
<td>00000</td>
<td>0052</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0053</td>
<td>0054</td>
<td>0055</td>
<td>00000</td>
<td>0056</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0057</td>
<td>0058</td>
<td>0059</td>
<td>00000</td>
<td>0060</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0061</td>
<td>0062</td>
<td>0063</td>
<td>00000</td>
<td>0064</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0065</td>
<td>0066</td>
<td>0067</td>
<td>00000</td>
<td>0068</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0069</td>
<td>0070</td>
<td>0071</td>
<td>00000</td>
<td>0072</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0073</td>
<td>0074</td>
<td>0075</td>
<td>00000</td>
<td>0076</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0077</td>
<td>0078</td>
<td>0079</td>
<td>00000</td>
<td>0080</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0081</td>
<td>0082</td>
<td>0083</td>
<td>00000</td>
<td>0084</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0085</td>
<td>0086</td>
<td>0087</td>
<td>00000</td>
<td>0088</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0089</td>
<td>0090</td>
<td>0091</td>
<td>00000</td>
<td>0092</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0093</td>
<td>0094</td>
<td>0095</td>
<td>00000</td>
<td>0096</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0097</td>
<td>0098</td>
<td>0099</td>
<td>00000</td>
<td>0100</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0101</td>
<td>0102</td>
<td>0103</td>
<td>00000</td>
<td>0104</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0105</td>
<td>0106</td>
<td>0107</td>
<td>00000</td>
<td>0108</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0109</td>
<td>0110</td>
<td>0111</td>
<td>00000</td>
<td>0112</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0113</td>
<td>0114</td>
<td>0115</td>
<td>00000</td>
<td>0116</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0117</td>
<td>0118</td>
<td>0119</td>
<td>00000</td>
<td>0120</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0121</td>
<td>0122</td>
<td>0123</td>
<td>00000</td>
<td>0124</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0125</td>
<td>0126</td>
<td>0127</td>
<td>00000</td>
<td>0128</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0129</td>
<td>0130</td>
<td>0131</td>
<td>00000</td>
<td>0132</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0133</td>
<td>0134</td>
<td>0135</td>
<td>00000</td>
<td>0136</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0137</td>
<td>0138</td>
<td>0139</td>
<td>00000</td>
<td>0140</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0141</td>
<td>0142</td>
<td>0143</td>
<td>00000</td>
<td>0144</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0145</td>
<td>0146</td>
<td>0147</td>
<td>00000</td>
<td>0148</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0149</td>
<td>0150</td>
<td>0151</td>
<td>00000</td>
<td>0152</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0153</td>
<td>0154</td>
<td>0155</td>
<td>00000</td>
<td>0156</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0157</td>
<td>0158</td>
<td>0159</td>
<td>00000</td>
<td>0160</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0161</td>
<td>0162</td>
<td>0163</td>
<td>00000</td>
<td>0164</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0165</td>
<td>0166</td>
<td>0167</td>
<td>00000</td>
<td>0168</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0169</td>
<td>0170</td>
<td>0171</td>
<td>00000</td>
<td>0172</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0173</td>
<td>0174</td>
<td>0175</td>
<td>00000</td>
<td>0176</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0177</td>
<td>0178</td>
<td>0179</td>
<td>00000</td>
<td>0180</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0181</td>
<td>0182</td>
<td>0183</td>
<td>00000</td>
<td>0184</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0185</td>
<td>0186</td>
<td>0187</td>
<td>00000</td>
<td>0188</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0189</td>
<td>0190</td>
<td>0191</td>
<td>00000</td>
<td>0192</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0193</td>
<td>0194</td>
<td>0195</td>
<td>00000</td>
<td>0196</td>
<td>00000</td>
</tr>
<tr>
<td>00000</td>
<td>0197</td>
<td>0198</td>
<td>0199</td>
<td>00000</td>
<td>0200</td>
<td>00000</td>
</tr>
</tbody>
</table>

Decimal and Hexadecimal Disk Addresses 131
This page intentionally left blank.
Appendix C. Data Formats

Disk System Format (DSF)

Unless otherwise instructed, the Disk Management Program (DMP) automatically converts programs in Card System Format (CDS) to Disk System Format (DSF), when programs are stored on disk. Likewise, programs in DSF are converted to CDS when dumping from to the 1442 card read punch. Disk System Format is shown in Figure 18; Card System Format is described elsewhere in this appendix.

Figure 18. Disk System Format

PROGRAM HEADER FORMAT

The contents of the program header record (see Figure 18) vary with the type of subroutine with which it is associated. The first 12 words of the program header record for the six...
types of programs are identified except for word 6, which is 9 less than the number of words in the program header record. The format of these 12 words is as follows:

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Zero</td>
</tr>
<tr>
<td>2</td>
<td>Checksum if source was cards, otherwise zero</td>
</tr>
<tr>
<td>3</td>
<td>Type</td>
</tr>
<tr>
<td>4</td>
<td>Word 4 of each program header will contain the length of INSKEL COMMON defined in the program (as supplied by the FORTRAN Compiler)</td>
</tr>
<tr>
<td>5</td>
<td>Length of coreload COMMON (words)</td>
</tr>
<tr>
<td>6</td>
<td>Length of program header record minus 9</td>
</tr>
<tr>
<td>7</td>
<td>Indicator bits for LIBFs in the FORTRAN I/O table</td>
</tr>
<tr>
<td>8</td>
<td>Length of program, including program header record, in disk blocks</td>
</tr>
<tr>
<td>9</td>
<td>Number of files defined</td>
</tr>
<tr>
<td>10-11</td>
<td>Name of entry point 1</td>
</tr>
<tr>
<td>12</td>
<td>Address of entry point 1 (absolute for type 1, relative to zero otherwise)</td>
</tr>
</tbody>
</table>

The indicator bits for word 7 are as follows:

<table>
<thead>
<tr>
<th>Bit Position</th>
<th>Indication</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>= 1 if 2311 subroutine DSOR or DSCR is required by FORTRAN I/O or assembler-language program I/O</td>
</tr>
<tr>
<td>1,2</td>
<td>= 00 if integer size unspecified</td>
</tr>
<tr>
<td></td>
<td>= 01 if single word integers</td>
</tr>
<tr>
<td></td>
<td>= 10 if multiple word integers</td>
</tr>
<tr>
<td>3,4</td>
<td>= 00 if precision unspecified</td>
</tr>
<tr>
<td></td>
<td>= 01 if standard precision</td>
</tr>
<tr>
<td></td>
<td>= 10 if extended precision</td>
</tr>
<tr>
<td>5</td>
<td>= 1 if TYPEN/EBPRT required by FORTRAN I/O</td>
</tr>
<tr>
<td>6</td>
<td>= 1 if CARDN/COBRT required by FORTRAN I/O</td>
</tr>
<tr>
<td>7</td>
<td>= 1 if PRNTN/EBPRT required by FORTRAN I/O</td>
</tr>
<tr>
<td>8</td>
<td>= 1 if PAPTN/PAPRT required by FORTRAN I/O</td>
</tr>
<tr>
<td>9</td>
<td>= 1 if MAGT required by FORTRAN I/O</td>
</tr>
<tr>
<td>10</td>
<td>= 1 if TYPEN/HOLEX required by FORTRAN I/O</td>
</tr>
<tr>
<td>11</td>
<td>= 1 if PLOTX required by FORTRAN I/O</td>
</tr>
<tr>
<td>12</td>
<td>= 1 if MPX compiled/assembled</td>
</tr>
<tr>
<td>13</td>
<td>= 1 if reentrant program</td>
</tr>
<tr>
<td>14</td>
<td>= 1 if unformatted disk I/O</td>
</tr>
<tr>
<td>15</td>
<td>= 1 if unformatted magnetic tape I/O</td>
</tr>
</tbody>
</table>

The indicators in bits 1-4 and 12-13 are required in all header types. Bits 5-11 and 14-15 are required only in type 2 headers.

After the first 12 words, the program header record format depends on the type of program. The header record for types 1 and 2 (absolute and relocatable mainline, respectively) consists of the first 12 words. The program types and their header record formats are shown below:

<table>
<thead>
<tr>
<th>Type Code</th>
<th>Type of Program</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Mainline (absolute)</td>
</tr>
<tr>
<td>2</td>
<td>Mainline (relocatable)</td>
</tr>
<tr>
<td>3</td>
<td>Subroutine, not an ISS, referenced by LIBF</td>
</tr>
<tr>
<td>4</td>
<td>Subroutine, not an ISS, referenced by CALL</td>
</tr>
<tr>
<td>5</td>
<td>Interrupt service subroutine (ISS) referenced by LIBF</td>
</tr>
<tr>
<td>6</td>
<td>Interrupt service subroutine (ISS) referenced by CALL</td>
</tr>
</tbody>
</table>
Program formats for type 3 and 4 programs:

<table>
<thead>
<tr>
<th>Words</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>13-14</td>
<td>Name of entry point 2 (30 bits, right-justified)</td>
</tr>
<tr>
<td>15</td>
<td>Address of entry point 2 (relative to zero)</td>
</tr>
<tr>
<td>16-17</td>
<td>Name of entry point 3 (30 bits, right-justified)</td>
</tr>
<tr>
<td>18</td>
<td>Address of entry point 3 (relative to zero)</td>
</tr>
<tr>
<td>19-54</td>
<td>Three words per entry point as above, to a maximum of 14 entry points. The header record ends at the last defined entry point; thus, it is of variable length.</td>
</tr>
</tbody>
</table>

Program formats for type 5 and 6 programs:

<table>
<thead>
<tr>
<th>Words</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>Contains zero</td>
</tr>
<tr>
<td>14</td>
<td>Contains the number of interrupt service entry points in the ISS</td>
</tr>
<tr>
<td>15</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

**Disk Data Format (DDF)**

Disk Data Format (DDF) describes the format of information placed in the Disk Fixed Area or in Batch-Processing Working Storage as a result of the DMP control statement *STOREDATA*. Disk Data Format consists of 320 binary words per sector; there are no headers, trailers, or indicator words.

**Card System Format (CDS)**

Words in columns 1 through 72 are punched in binary form, 1 1/3 columns for each word. Each card may contain 54 words in binary form. Columns 73 through 80 may contain card identification or sequence numbers, in IBM card code. Data cards are punched with up to 45 words in columns 13 through 72. See Figure 19.

**MAINLINE HEADER CARD**

A mainline header card contains the size of the common area and the size of the Batch-processing Working Storage. It is the first card of the mainline program. Its format is as follows:

<table>
<thead>
<tr>
<th>Words</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reserved</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Type code (first 8 bits):</td>
</tr>
<tr>
<td></td>
<td>0000 0001 - absolute</td>
</tr>
<tr>
<td></td>
<td>0000 0010 - relocatable</td>
</tr>
<tr>
<td>4</td>
<td>Word 4 of each mainline header card will contain the length of INSKEI COMMON defined in the program (as supplied by the FORTRAN Compiler)</td>
</tr>
<tr>
<td>5</td>
<td>Length of COMMON storage area (FORTRAN mainline program only)</td>
</tr>
</tbody>
</table>
The checksum in word 2 is the twos complement of the logical sum of the record count (position of the record within the deck) and the data word(s). The logical sum is obtained by summing the data word(s) and the record count arithmetically with the addition of a one each time a carry occurs out of the high-order position of the A-register.

**DATA CARDS**

Data cards contain the instructions and data that constitute the assembled program. The format is as follows:

<table>
<thead>
<tr>
<th>Words</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Location. (The relative load address of the first data word of the card or record. Succeeding words go into higher-numbered main-storage locations. The relocation factor must be added to this address to obtain the actual load address. For an absolute program the relocation factor is zero.)</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Type code (first 8 bits): 0000 1010 Data word count (last 8 bits)</td>
</tr>
<tr>
<td>4-9</td>
<td>Relocation indicators (2 bits per data word): 00 - nonrelocatable or absolute 01 - relocatable 10 - LIBF (one-word call) 11 - CALL (two-word call)</td>
</tr>
<tr>
<td>10</td>
<td>Data word 1</td>
</tr>
<tr>
<td>11-54</td>
<td>Data words 2 through 45</td>
</tr>
</tbody>
</table>

**EOP CARD**

An end of program card (EOP) is the last card of each program and subroutine. The format is as follows:

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Starting location of next subroutine (this number is always even and is assigned by the Macro Assembler)</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Type code (first 8 bits): 0000 1111 Last 8 bits: 0000 0000</td>
</tr>
<tr>
<td>4</td>
<td>XEQ address, if mainline program</td>
</tr>
<tr>
<td>5-54</td>
<td>Reserved</td>
</tr>
</tbody>
</table>
SUBROUTINE HEADER CARD

A maximum of 14 entry points can be defined for each subroutine. The format of the subroutine header card is as follows:

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reserved</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Type code (first 8 bits):</td>
</tr>
<tr>
<td></td>
<td>0000 0011 - to be called by a one-word call only (LIBF)</td>
</tr>
<tr>
<td></td>
<td>0000 0100 - to be called by a two-word call only (CALL)</td>
</tr>
<tr>
<td>4-5</td>
<td>Reserved</td>
</tr>
<tr>
<td>6</td>
<td>Number of entry points times three</td>
</tr>
<tr>
<td>7</td>
<td>See Program Header Format, word 7</td>
</tr>
<tr>
<td>8</td>
<td>Program length</td>
</tr>
<tr>
<td>9</td>
<td>Reserved</td>
</tr>
<tr>
<td>10-11</td>
<td>Name of entry point 1</td>
</tr>
<tr>
<td>12</td>
<td>Relative address of entry point 1</td>
</tr>
<tr>
<td>13-51</td>
<td>Names and relative addresses of entry points 2 through 14</td>
</tr>
<tr>
<td>52-64</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

ISS HEADER CARD

An ISS (interrupt servicing subroutine) header card for each interrupt servicing subroutine identifies the entry point defined by an ISS statement. Only one entry point can be defined for each subroutine. The format of the ISS header card is as follows:

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reserved</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Type code (first 8 bits):</td>
</tr>
<tr>
<td></td>
<td>0000 0101 - to be called by a one-word call only (LIBF)</td>
</tr>
<tr>
<td></td>
<td>0000 0110 - to be called by a two-word call only (CALL)</td>
</tr>
<tr>
<td></td>
<td>Precision code (last 8 bits):</td>
</tr>
<tr>
<td></td>
<td>0000 0000 - undefined</td>
</tr>
<tr>
<td></td>
<td>0000 0001 - standard</td>
</tr>
<tr>
<td></td>
<td>0000 0010 - extended</td>
</tr>
<tr>
<td>4-5</td>
<td>Reserved</td>
</tr>
<tr>
<td>6</td>
<td>Six</td>
</tr>
<tr>
<td>7</td>
<td>See Program Header Format, word 7</td>
</tr>
<tr>
<td>8-9</td>
<td>Reserved</td>
</tr>
<tr>
<td>10-11</td>
<td>Subroutine name</td>
</tr>
<tr>
<td>12</td>
<td>Relative entry address</td>
</tr>
<tr>
<td>13</td>
<td>Zero</td>
</tr>
<tr>
<td>14</td>
<td>The number of interrupt service entry points in the ISS</td>
</tr>
<tr>
<td>15-54</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

SECTOR BREAK CARDS

Sector break cards are binary cards used by the System Loader and DMON function of DMP to cause programs or phases of programs to start loading at the beginning of a sector. Sequence numbers are punched in columns 73-80 in IBM Card Code. The sector break cards are identified by a 1 punch in column 4 (binary word 3).

A Type 1 sector break card indicates to the System Loader or DMON function of DMP that the following program or phase should start loading at the next new sector address available.

The format of the sector break card is as follows:

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reserved</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Type code (first 8 bits): 0000 0001</td>
</tr>
<tr>
<td>4-9</td>
<td>Reserved</td>
</tr>
<tr>
<td>10-11</td>
<td>Phase name</td>
</tr>
<tr>
<td>12-54</td>
<td>Reserved</td>
</tr>
</tbody>
</table>
Card Data Format (CDD)

All words are punched in binary form, 1 1/3 columns for each word. Each card may contain 60 words. Data cards are punched with up to 56 words in columns 5 through 79. See Figure 19.

![Card Data Format](image)

Figure 19. Card Data Format

DISK DUMP FOR PHYSICAL OR MAPPED 1810s

The output from the DUMP function of DMP or the Disk-Dump-to-Cards Utility Program (BDUMP) is in Card Data Format. The 1810 dumped disk data is punched into cards as follows:

**CYLINDER BREAK CARD FORMAT**

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/FFFF</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Reserved</td>
</tr>
<tr>
<td>4</td>
<td>Sector address</td>
</tr>
<tr>
<td>5</td>
<td>File-protection status</td>
</tr>
<tr>
<td>6-59</td>
<td>Reserved</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>

**DATA CARD FORMAT**

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Relative address of data on cylinder</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Word count (number of data words)</td>
</tr>
<tr>
<td>4-59</td>
<td>Reserved</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>
DISK DUMP FOR 2311s

The output from the 2311 Disk Dump and Reload Program (BDCRL) is in Card Data Format. A card deck from a 2311 dump can be identified by the /FFFF in word 5 of the first cylinder break card. Each new record begins on a new card. The first word of these new record cards contains zero. The format of the other cards is as follows:

FIRST CYLINDER BREAK CARD

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/FFFF</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Reserved</td>
</tr>
<tr>
<td>4</td>
<td>Cylinder address</td>
</tr>
<tr>
<td>5</td>
<td>/FFFF</td>
</tr>
<tr>
<td>6-59</td>
<td>Reserved</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>

SUBSEQUENT CYLINDER BREAK CARDS

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/FFFF</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Reserved</td>
</tr>
<tr>
<td>4</td>
<td>Cylinder address</td>
</tr>
<tr>
<td>5-59</td>
<td>Reserved</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>

CYLINDER DATA CARD

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Data word address, relative to the beginning of the record</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Word count (number of data words)</td>
</tr>
<tr>
<td>4-59</td>
<td>Data words</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>

END-OF-ALL-CYLINDER CARDS

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reserved</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>/FFFF</td>
</tr>
<tr>
<td>4-59</td>
<td>Reserved</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>
The output of the Dump-to-Cards utility program is in binary format, 3/4 of a 16-bit binary word per card column. This format allows 56 data words to be punched in each 80-column card, where 1-1/3 columns equal one binary word. The main storage words are output to cards as follows:

**FIRST CARD FORMAT**

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/FFFF</td>
</tr>
<tr>
<td>3</td>
<td>Starting address of dump program</td>
</tr>
<tr>
<td>4</td>
<td>Ending address of dump program</td>
</tr>
</tbody>
</table>

**DATA CARD FORMAT**

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Relative address of data in main storage</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>Word count (number of data words)</td>
</tr>
<tr>
<td>4-59</td>
<td>Data words</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>

**LAST CARD FORMAT**

<table>
<thead>
<tr>
<th>Word</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Zero</td>
</tr>
<tr>
<td>2</td>
<td>Checksum</td>
</tr>
<tr>
<td>3</td>
<td>/FFFF</td>
</tr>
<tr>
<td>4</td>
<td>Execution Address</td>
</tr>
<tr>
<td>5-59</td>
<td>Not used</td>
</tr>
<tr>
<td>60</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>

**Print Data Format (PRD)**

Print Data Format is the format in which programs and data files are dumped to the print devices. The dump is printed by sector. The dump of each sector consists of a heading followed by the data on the sector dumped.

The heading for each sector dumped consists of the absolute sector address of the sector being dumped and the location of the first data word in the sector dumped relative to the start of the program or data file being dumped.

The body of the dump of a sector consists of from one to twenty lines, each line containing the location of the first data word in the line printed relative to the start of the sector being dumped, followed by 16 hexadecimal data words. Two spaces separate each data word from those adjacent to it.

The printing of lines that contain exactly the same data as the previous line printed is suppressed. One line of blanks is printed to indicate that lines have been suppressed. An exception to the procedure may occur in the printing of the last line of the dump of a sector; the last line is always printed.
Appendix D. MPX Sample Program

The MPX sample program contains five background-processing coreloads written in FORTRAN. The sample program deck is punched during system generation and is run as a background-processing job. System assignments must be made before the sample program can be run.

Operating Procedure

Enter the logical unit number of the input and output device in the source cards as follows. (SAMxxxxx is the source card ID.)

\[ I = \text{SAM00220} \]

Set I equal to the LUN of the device which will be used to input user data during the program run. Keyboard input is recommended.

\[ J = \text{SAM00250} \]

Set J equal to the LUN of the output device used for listing purposes. List printer is recommended.

Note: If there is no 1443 on the system, it is advisable to remove the *LIST ALL cards from the source deck.

Run the sample program as a batch-processing job.

The source deck is compiled and listed on the list printer. At the same time, operating instructions are printed on the system printer. The system and list printer printouts (these can be the same printer) are listed below. The background-processing coreloads are then built, executed, and finally deleted.

Additional operating procedures are shown on the system printer printout.
IBM 1800 OACS  MPX/DSM  00,000 00 JAN 00
SEN SW 0 ON ABSOLUTE LOADER
SEN SW 1 ON LOAD BP MONITOR
SEN SW 2 ON SET CLOCK VIA DATA SW
SEN SW 3 ON SET DATE VIA DATA SW

// JOB 00 JAN 00 00,000 HRS

// MPX SAMPLE PROGRAM
// SAMPLE PROBLEM READS AI POINTS AND OUTPUTS READING ON USER
// SPECIFIED DEVICE. FORTRAN LOGICAL UNIT NUMBERS FOR INPUT
// AND OUTPUT DEVICES ARE SPECIFIED IN THE FIRST PROGRAM (SAMPL)
// THE USER MUST PUNCH IN THE VALUES FOR THESE DEVICES.
// RELAY INPUT POINTS ARE 00000 - 04005.
// 52 POINTS ARE 04096 - 08101, AFTER POINTS ARE READ AND
// VALUES OUTPUT, A VALUE IS ENTERED VIA A USER SPECIFIED
// DEVICE. FORMAT 15, THE SQUARE ROOT OF THIS VALUE
// IS CALCULATED AND OUTPUT ON A USER SPECIFIED DEVICE.
// JOB 00 JAN 00 00,000 HRS
// FOR SAMPL 00 JAN 00 00,026 HRS
END OF COMPILATION

SAMP
DMP FUNCTION COMPLETED
// DMP
+STORECIL SAMP SAMPL
*CCEND
MPX, BUILD SAMPL

MPX, SAMPL LD XQ

045 CORELDS NOT END
SAMP
CL WC OF 0048 STORED AT 04C0
DMP FUNCTION COMPLETED
// FOR SAMPZ 00 JAN 00 00,052 HRS
END OF COMPILATION

SAMPZ
DMP FUNCTION COMPLETED
// DMP
+STORECIL SAMPZ SAMPZ
*CCEND
MPX, BUILD SAMPZ

MPX, SAMPZ LD XQ

CL WC OF 0048 STORED AT 04C1
DMP FUNCTION COMPLETED
// FOR SAMPY 00 JAN 00 00,088 HRS
END OF COMPILATION

SAMPY
DMP FUNCTION COMPLETED
// DMP
+STORECIL SAMPY SAMPY
*CCEND
MPX, BUILD SAMPY

MPX, SAMPY LD XQ

CL WC OF 004E STORED AT 04C2
DMP FUNCTION COMPLETED
// FOR SAMPC 00 JAN 00 00,124 HRS
END OF COMPILATION

SAMPC
DMP FUNCTION COMPLETED
// DMP
+STORECIL SAMPC SAMPC
*CCEND
MPX, BUILD SAMPC

MPX, SAMPC LD XQ

D45 CORELDS NOT FND
SAMP
CL WC OF 00BE STORED AT 04C3
DMP FUNCTION COMPLETED
// FOR SAMPA 00 JAN 00 00,160 HRS
END OF COMPILATION

SAMPA
DMP FUNCTION COMPLETED
// DMP
+STORECIL SAMPA SAMPA
*CCEND
MPX, BUILD SAMPA

MPX, SAMPA LD XQ

CL WC OF 04C4 STORED AT 04C4
DMP FUNCTION COMPLETED
// FOR SAMPB 00 JAN 00 00,189 HRS
END OF COMPILATION

SAMPB
DMP FUNCTION COMPLETED
// DMP
+STORECIL SAMPB SAMPB
*CCEND
MPX, BUILD SAMPB

MPX, SAMPB LD XQ

CL WC OF 04C4 STORED AT 04C4
DMP FUNCTION COMPLETED
// FOR SAMPA 00 JAN 00 00,189 HRS
END OF COMPILATION

SAMPA
DMP FUNCTION COMPLETED
// DMP
+STORECIL SAMPA SAMPA
*CCEND
MPX, BUILD SAMPA

142  1800 MPX Operating Procedures
MPX Sample Program

MPX Sample Program

COMMON I,J,L,K,A
C START OF PROGRAM
C DEFINE LOGICAL UNIT NUMBER OF INPUT DEVICE.
I=96
C DEFINE LOGICAL UNIT NUMBER OF OUTPUT DEVICE.
J=5
CALL LINK(SAMPA)
END

VARIABLE ALLOCATIONS
I(IIC)=FFFF J(JIC)=FFFF L(LIC)=FFFF K(KIC)=FFFF A(AIC)=FFFF

FEATURES SUPPORTED
NONPROCESS
ONE WORD INTEGERS

INTEGER CONSTANTS
6=0000 3=0001

CORE REQUIREMENTS FOR SAMPL
COMMON & INSKEL COMMON O VARIABLES O PROGRAM 14

END OF COMPILATION

SAMPL
DMP FUNCTION COMPLETED
// DMP
*STORECIL SAMPL SAMPL
*CCEND

~

APDX D
MPX, BUILD SAMPL

CORE LOAD MAP

TYPE NAME ARG1 ARG2

*COW TABLE 8002 0012
*FRI TABLE 8014 001E
*GMT TABLE 8032 0008
*MAIN SAMPL 8038
*CLNY SAMPL 8038
*CLNT SAMPL 8038
*CORE ROMA 7F80
*COMM FPPA 0006
*MPX, SAMPL LOCK 0X

185 CURIOUS NOT PNO SAMPL CORE LOAD MAP

TYPE NAME ARG1 ARG2

*COW TABLE 8002 0012
*FRI TABLE 8014 001E
*GMT TABLE 8032 0008
*MAIN SAMPL 8038
*CLNT SAMPL 8038
*CLNY SAMPL 8038
*CORE ROMA 7F80
*COMM FPPA 0006
*MPX, SAMPL LO X0

MPX, BUILD SAMPL

CORE LOAD MAP

TYPE NAME ARG1 ARG2

*COW TABLE 8002 0012
*FRI TABLE 8014 001E
*GMT TABLE 8032 0008
*MAIN SAMPL 8038
*CLNY SAMPL 8038
*CLNT SAMPL 8038
*CORE ROMA 7F80
*COMM FPPA 0006

MPX BUILD SAMPL

CORE LOAD MAP

TYPE NAME ARG1 ARG2

*COW TABLE 8002 0012
*FRI TABLE 8014 001E
*GMT TABLE 8032 0008
*MAIN SAMPL 8038
*CLNY SAMPL 8038
*CLNT SAMPL 8038
*CORE ROMA 7F80

144 1800 MPX Operating Procedures
CL WC DF O0CB STORED AT 04C1
UMP FUNCTION COMPLETED
// FOR SAMPY 00 JAN 00 00.088 HRS
## MPX SAMPLE PROGRAM
*LIST ALL
ONE WORD INTEGERS
#NONPROCESS PROGRAM
*IGCS(10CSKEYBAORD,TYPEWRITER,1443 PRINTER,CARD)

// FOR SAMY 00 JAN 00 00.10 MRS
** MPX SAMPLE PROGRAM
LIST ALL
ONE WORD INTEGERS
NONPROCESS PROGRAM
10CS, TYPEWRITER, 1443 PRINTER/CARD

VARIABLE ALLOCATIONS
J(1:J,1:K,A)
COMMON (I, J, L, K, A)
WRITE(I, J)
7 FORMAT(I, ENTER NUMBER--FORMAT 15*)
READ(I, J, K)
AMK
CALL LINK (SAMPY)
END

UNREFERENCED STATEMENTS
6
STATEMENT ALLOCATIONS
7 =0000 2 =0001 6 =0010

FEATURES SUPPORTED
NONPROCESS
ONE WORD INTEGERS
IOCS

MINALLY SUPPORTED
FSTO
FLOAT
MRED
MWRT
MCOMP
MI01
TYPEN
HOLE
PRNTN
EBPRT
CARON

CORE REQUIREMENTS FOR SAMPY
COMMON 6 INSKEL COMMON 0
VARIABLES 0 PROGRAM 36

SAMYPY
UMP FUNCTION COMPLETED
// OMP
*STORECIL SAMYPY SAMYPY
*CCEND
MPG BUILD SAMYPY
CORE LOAD MAP

MPX, BUILD SAMYPY

CORE LOAD MAP
TYPE NAME AARG AARG
*COW TABLE 8002 0012
+FJU TABLE 8014 001E
+FVS TABLE 8032 001B
+CMNT TABLE 80BE 0008
MAIN SAMYPY 8086
CLNT SAMYPY 8050
CALL EBPR 807A
CALL M01 802A
CALL M02 8039
LIBF MWRT 8056 8032 R
LIBF MCOMP 8031 8038 R
LIBF MRED 8035 8033 R
LIBF MI01 8036 803B
LIBF MSTD 8038 803E
LIBF FSTD 8031 8041
CALL SAMYPY 8054
CALL EPRT 8058
CALL FSTD 8045
CALL SRCAL 8023
CALL IDU 8016 R
CALL IPRT 8030 R
CALL GETBF 8063
CALL FPRT 8055
CALL IDX 8009
LIBF ADMK 8040 8044 R
LIBF IPRT 8010 8047
LIBF IOUX 8046 804A
LIBF FSTD 8031 8041
CALL FREUP 8030
CORE 80OD 0054
COMMON 8004 0216
COMMON 8000 0066
COMMON 8000 0044 R
COMMON 8000 0030 R
MPX, SAMYPY LO KU
CL WC OF O0CE STORED AT 04CD
UMP FUNCTION COMPLETED
// FOR SAMPC 00 JAN 00 00.124 MRS
## MPX SAMPLE PROGRAM
*LIST ALL
ONE WORD INTEGERS
#NONPROCESS PROGRAM
*IGCS(10CSKEYBAORD,TYPEWRITER,1443 PRINTER)

// FOR SAMPC 00 JAN 00 00.124 MRS
## MPX SAMPLE PROGRAM
*LIST ALL
ONE WORD INTEGERS
#NONPROCESS PROGRAM
*IGCS(10CSKEYBAORD,TYPEWRITER,1443 PRINTER)

COMMON (I, J, L, K, A)
WRITE(I, J, K)
5 FORMAT(I, ENTER NUMBER--FORMAT 15*)
READ(I, J, K)
AMK
CALL LINK (SAMPY)
END

MPX Sample Program 145
VARIABLE ALLOCATIONS
111C1=FFFF   JIIC/=FFFE   KIICI=FFEC   AIRC/=FFFA   ITEST/=0000

UNREFERENCED STATEMENTS
4

STATEMENT ALLOCATIONS
5  =0004  4  =0019  10  =002A  6  =002F

FEATURES SUPPORTED
NONPROCESS
ONE WORD INTEGERS
IDCS

CALLED SUBPROGRAMS
DATSM CONGO MRT MCMP SIG1 PAUSE TYPEN PNTIN EBPR

INTEGER CONSTANTS
0=0004  0=0005

STATEMENT ALLOCATIONS
4  =0004  4  =0019  10  =002A  6  =002F

END OF COMPILATION

SAMPC
DCMP FUNCTION COMPLETED
// UNF

#STIMECL   SAMPC   SAMPC

#CCEND

MPA, BUILD SAMPC

CORE LOAD MAP

TYPE NAME ARG1 ARG2

=OEC TABLE 8002 0012
=ID TABLE 8014 001E
=IVY TABLE 8032 001B
=CM TABLE NONE 000C
MAIN SAMPC 8073
CLNT SAMPC 8090
CALL EBPR HORE
LIBF MRT 830C 0032 R
LIBF MGO 8374 0035
LIBF MCMP 8357 0038
LIBF PAUSE 8850 003B R
CALL GATSM 8854
LIBF CPGU 8879 003E R
CLNT SAMPC 8045
CLNT SAMPC 809A
CALL PRF 8880
CALL PENG 888F R
CALL SBBAL 892R
CALL ICU 80AA R
CALL IPFX 80AB R
CALL GETRF 8939
CALL PREBF 892D
CALL PDXF 8931
LIBF ADCK RABR 8041 R
LIBF FLOAT 804B 0044
LIBF IFIX 804C 0047
CALL FREBF 892D
LIBF NRM 8042 0044
FLIB 804E 0054
CLNT 804E 736A
COMM 804F 0006
MPA, SAMPC LD XU

END CORELS NOT FND

SAMPC
CL WC IF FREE STORED AT 0409
DCMP FUNCTION COMPLETED
// FIN SAMPC 00 JAN 00 00.160 HRS

** MPX SAMPLE PROGRAM

*LIST ALL

ONE WORD INTEGERS

*NONPR0CESS PROGRAM

DIMENSION LIST('IARA(3)
COMMON 'I,J,L,K
EQUIVALENCE (I)TEST,LST(4))
DATA (LIST(3) = 2009.0,1.12,4.13,1.14,2.15)
IARA(3) = K
4 CALL ATSLIST(I101)
1 IF (LIST(101)=2)
2 L=IARA(1)
3 GO TO 154,4,4,4,4,11,TEST
END

VARIABLE ALLOCATIONS
111C1=FFFF   JIIC/=FFFE   KIICI=FFEC   AIRC/=FFFA

STATEMENT ALLOCATIONS
5  =0004  4  =0019  10  =002A  6  =002F

146  1800 MPX Operating Procedures
FEATURES SUPPORTED
NONPROCES
ONE WORD INTEGERS

CALLED SUBPROGRAMS
  IADOR
  AISN
  COMGO
  ISTOX

CORE REQUIREMENTS FOR SAMPB
COMMON A INSKEL COMMON 0
VARIABLES 16 PROGRAM 50

END OF COMPILATION
SAMPB
DMP FUNCTION COMPLETED
// DMP SAMPB SAMPB
STORECIL SAMPB SAMPB
CCEND

MPX, RUNJO SAMPB

CORE LOAD MAP
TYPE NAME ARG1 ARG2

#COW TABLE RO02 0012
#ED TABLE RO14 0016
#VT TABLE RO32 0000
#CT TABLE RO3C 0000
MAIN SAMPB 0054
CLNT SAMPB 008E
LIBF ISTOX 0082 0032 R
LIBF AISN 0082 R
LIBF COMGO 0267 8035 R
CLNT SAMPB 8042
LIBF ADRCK 80EA 0038 R
LIBF ABRTQ 804E R
LIBF FERQ 803F R
LIBF PREUP 80BC R
CORE 006A 7834
COMM FFPA 0006
MPX, SAMPB LU XQ

CL WC OF DC4 STORPD AT 04F3
DMP FUNCTION COMPLETED
// FOR SAMPB 00 JAN 00 00.189 MRS

**MPX SAMPLE PROGRAM
NONWORD INTEGERS
NONPROCESS PROGRAM

COMMON I,J,K,A
10 WRITE (J,11)
1 FORMAT(' ENTER A POINT TO BE READ-- FORMAT 15')
READ(I,2)
2 FORMAT(15)
CALL LINK(SAMPB)
END

VARIABLE ALLOCATIONS
1(IC)=FFFF J(IC)=FFFE L(IC)=FFFP K(IC)=FFFC A(RC)=FFFA

UNREFERENCED STATEMENTS
10

STATEMENT ALLOCATIONS
1 =0000 2 =0015 10 =0017

FEATURES SUPPORTED
NONPROCES
ONE WORD INTEGERS

CALLED SUBPROGRAMS

CORE REQUIREMENTS FOR SAMPB
COMMON A INSKEL COMMON 0
VARIABLES 16 PROGRAM 38

END OF COMPILATION
SAMPB
DMP FUNCTION COMPLETED
// DMP SAMPB SAMPB
STORECIL SAMPB SAMPB
CCEND
### MPX, BUILD SAIIPA

<table>
<thead>
<tr>
<th>CORE LOAD MAP</th>
<th>TYPE NAME</th>
<th>ARG1</th>
<th>ARG2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MAP</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **COW** TABLE 8002 0012
- **F10** TABLE 8014 001E
- **TVT** TABLE 8032 0018
- **CMT** TABLE 804A 0008

**MAIN** SAIIPA 8069
**CLNT** SAIIPA 804C
**CALL** EPRT 8078
**CALL** CARDS 8122
**CALL** HOLER 82AC
**CALL** HOLE% 8307
**LIBF** MWRT 8504 8032
**LIBF** MCOMP 861F 8035
**LIBF** MRED 8500 8038
**LIBF** M101 863C 8038
**CALL** PRT 8AF8
**CALL** FEROR 8845
**CALL** SRCAL 8C73
**CALL** IOU 8C76
**CALL** IOFIX 8C00
**CALL** GETBF 8883
**CALL** FREBF 8875
**LIBF** ADRCK 8000 803E
**LIBF** FLOAT 8080 8041
**LIBF** IFIX 8DC4 8044
**LIBF** NWRT 8E34 0054
**CORE** 8E88 7172
**COMM** FFFA 0006

**MPX, SAIIPA LD XQ**
**CL WC OF 0E32 STORED AT 04E7**
**IMP FUNCTION COMPLETED**

```
// XEQ SAMPL FXO 00 JAN 00 00.568 HRS SAM01180
-10392
DAT SW 0 ON TO READ ANOTHER POINT
ENTER NUMBER--FORMAT 15
SQUARE ROOT OF 5 IS 0.223606859E 01
// JOB 00 JAN 00 00.402 HRS
// DMP 00 JAN 00 00.403 HRS
•DELET SAHPZ SAM01210
DMP FUNCTION COMPLETED
•DELET SAMPY SAM01220
DMP FUNCTION COMPLETED
•DELET SAMPA SAM01230
DMP FUNCTION COMPLETED
•DELET SAMPR SAM01240
DMP FUNCTION COMPLETED
•DELET SAMPL SAM01250
DMP FUNCTION COMPLETED
```

```
// JOB 00 JAN 00 00.402 HRS
// DMP 00 JAN 00 00.403 HRS
•DUMPLET
LET PACK LABEL
```

### 1800 MPX Operating Procedures
IMP FUNCTION COMPLETED

END OF SAMPLE PROGRAM.

00 JAN 00 00.636 HRS 99999999

CL WC OF 03E2 STORED AT 04E7
DMP FUNCTION COMPLETED
// XEQ SAMPL F30 00 JAN 00 00.225 HRS

SA01180

00002
00005
// JOB 00 JAN 00 00.374 HRS SA01190
// DMP 00 JAN 00 00.375 HRS SA01200
*DELETE SAMPL SA01210
DMP FUNCTION COMPLETED
*DELETE SAMP2 SA01220
DMP FUNCTION COMPLETED
**DELETE SAMPY SA01230
DMP FUNCTION COMPLETED
**DELETE SAMPB SAM01240
DMP FUNCTION COMPLETED
**DELETE SAMPL SAM01250
DMP FUNCTION COMPLETED
// JOB 00 JAN 00 00.401 HRS SAM01260
// DMP 00 JAN 00 00.403 HRS SAM01270
*DUMPLET SAM01280
DMP FUNCTION COMPLETED
// END END OF SAMPLE PROGRAM. 00 JAN 00 00.417 HRS 99999999

IBM 1800 DACS HPM/BOM 00.419 00 JAN 00
SEN SW 0 ON ABSOLUTE LOADER
SEN SW 1 ON LOAD BP MONITOR
SEN SW 2 ON SET CLOCK VIA DATA SWS
SEN SW 3 ON SET DATE VIA DATA SWS

1800 MPX Operating Procedures
## Appendix E. Summary of Console Sense, Program, and Data Switch Operations

<table>
<thead>
<tr>
<th>Program</th>
<th>Console Switch Settings</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sense and Program</td>
<td>Data</td>
</tr>
<tr>
<td>BDUMP</td>
<td>All</td>
<td>First and last cylinder address</td>
</tr>
<tr>
<td>BDUPL</td>
<td>4-7 and 12-15 and 0-ON</td>
<td>Drive selection</td>
</tr>
<tr>
<td></td>
<td>1-ON</td>
<td>2311 copy</td>
</tr>
<tr>
<td>BDWAP</td>
<td>All</td>
<td>Number of tries</td>
</tr>
<tr>
<td></td>
<td>0-OFF</td>
<td>Reselect drive</td>
</tr>
<tr>
<td></td>
<td>15-ON</td>
<td>Select drive 1</td>
</tr>
<tr>
<td></td>
<td>14-ON</td>
<td>Select drive 2</td>
</tr>
<tr>
<td>BLIST</td>
<td>0-ON</td>
<td>List cards</td>
</tr>
<tr>
<td></td>
<td>1-ON</td>
<td>Sequence check</td>
</tr>
<tr>
<td>BOM</td>
<td>Absolute load</td>
<td>Execute or load to disk</td>
</tr>
<tr>
<td></td>
<td>Utilities 15-ON</td>
<td>Select utility function</td>
</tr>
<tr>
<td></td>
<td>Utilities 14-ON</td>
<td>Select printer code</td>
</tr>
<tr>
<td></td>
<td>Utilities 0-ON</td>
<td>Sequential sectors</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Disk dump function)</td>
</tr>
<tr>
<td></td>
<td>BDMON</td>
<td>Console interrupt to get Supervisor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Keyboard request to get Supervisor</td>
</tr>
<tr>
<td>CA Trace</td>
<td>15-ON</td>
<td>Suppress CA trace</td>
</tr>
<tr>
<td>CE Coreload and 1442 Diagnostic</td>
<td></td>
<td>Uses only CE switches</td>
</tr>
<tr>
<td>Check/Stop Trace</td>
<td>All</td>
<td>Select trace address</td>
</tr>
<tr>
<td>Cold Start (To 2311)</td>
<td>0-ON</td>
<td>Cold start on 2311</td>
</tr>
<tr>
<td></td>
<td>4-7</td>
<td>Mapped 1810 drive number</td>
</tr>
<tr>
<td></td>
<td>8-11</td>
<td>Control unit address</td>
</tr>
<tr>
<td></td>
<td>12-15</td>
<td>Physical 2311 which contains the system pack</td>
</tr>
<tr>
<td></td>
<td>All</td>
<td>Enter time</td>
</tr>
<tr>
<td></td>
<td>All</td>
<td>Enter date</td>
</tr>
<tr>
<td>Cold Start (To 1810)</td>
<td>15-ON</td>
<td>Executive on drive 1</td>
</tr>
<tr>
<td></td>
<td>14-ON</td>
<td>Executive on drive 2</td>
</tr>
<tr>
<td></td>
<td>All</td>
<td>Enter time</td>
</tr>
<tr>
<td></td>
<td>All</td>
<td>Enter date</td>
</tr>
<tr>
<td>Dump Analysis</td>
<td>15-ON</td>
<td>Proceed following checksum error</td>
</tr>
<tr>
<td>EAC</td>
<td>6-ON</td>
<td>HALT ON ERROR</td>
</tr>
<tr>
<td>FORTRAN</td>
<td>2-ON</td>
<td>Suppress listing</td>
</tr>
<tr>
<td>Full Trace</td>
<td>16-ON</td>
<td>Enable trace</td>
</tr>
<tr>
<td></td>
<td>All</td>
<td>Set limits</td>
</tr>
<tr>
<td></td>
<td>15-ON</td>
<td>Suppress print</td>
</tr>
<tr>
<td>Macro Assembler</td>
<td>2-ON</td>
<td>Suppress listing</td>
</tr>
<tr>
<td>Monitor trace</td>
<td>All</td>
<td>Select trace address</td>
</tr>
<tr>
<td>Supervisor</td>
<td>4-ON</td>
<td>Bypass not-ready 1443 printer</td>
</tr>
<tr>
<td>Trace define limit</td>
<td>5-ON</td>
<td>Console interrupt to select trace limit</td>
</tr>
<tr>
<td></td>
<td>7-OFF</td>
<td></td>
</tr>
<tr>
<td>*MON</td>
<td>1-ON</td>
<td>User option for over or underflow of system program</td>
</tr>
<tr>
<td></td>
<td>0-ON</td>
<td>Abort load in conjunction with switch 1 option</td>
</tr>
<tr>
<td></td>
<td>15-OFF</td>
<td>Print each new word</td>
</tr>
<tr>
<td></td>
<td>14-OFF</td>
<td>Suppress all MON printing but error messages</td>
</tr>
<tr>
<td>*SRFLE</td>
<td>15-ON</td>
<td>Suppress list or punch</td>
</tr>
</tbody>
</table>

Summary of Console Sense, Program, and Data Switch Operations 150.1
Glossary-Index

This section consists of a glossary merged with an index. The only MPX control statements included in the glossary-index are those for which the format and use are explained in this manual.

// SET 64
*DEFINE NDISK 45
*DLABL 45
*END BDPIP 82
*END DPIP 101
*END MLIST 102
*GETALT 82-83
*INITLZ 82-83
*MON 67-76

deleting system programs, 73-74
format, 68-69
installing PTFs, 72
loading new versions of programs, 74-76
patching disk words, 69-70
patching system programs, 71
printing PTF log, 72-73
replacing subroutines, 75-76
updating version/modification, 73

Absolute Loader 59,77
executing BDCRL, 78
executing BDPIP, 84
executing BDUMP, 79
executing BDUPL, 85
executing BDWAP, 86
executing BLIST, 88
executing BRELD, 88

Accumulator
use with cold start messages, 57-58
ADRCK
assembly, 40,42.2
equate cards, 36,39
Altering Main Storage 16
Altering System Programs 71
Assembler
See Macro Assembler.
Assignment Cards
format, 127-129
loading, 44
purpose, 17
use, 18

Background Processing 63-65. In a real-time system, the sequential execution of programs, usually scientific or data-processing in nature, done in VCORE during the time not needed for real-time tasks. These are kinds of programs that can be executed in a batch-processing system.

Basic Level. The level on which some programs, including background-processing programs, are executed in VCORE. This level is lower in precedence than any interrupt level.

Basic Operating Monitor (BOM). The set of programs and subroutines that direct the operation of a batch-processing MPX system. Functions of BOM are incorporated into the Executive in a real-time system.

Building, 46-47
card utilities, 77-89
cold start, 52
equate cards, 17,30,34-35,36-37
high core loader, 19-20
initializing, 63
loading, 44
reload, 59-60,91-92
system generation, 17
utility package, 91-102

Batch-Processing Monitor (BPMON) 48.
A set of programs that carry out batch-processing and background-processing operations; the Supervisor, FORTRAN Compiler, Macro Assembler, Builder, and Disk Management Program.
cold start, 51-52

Batch-Processing Monitor Supervisor (SUP), A program that directs all batch-processing and background-processing operations.
input to, 63-65
loading, 63
reload, 59-60
system generation, 17,46

Batch-Processing System, An MPX system that has no real-time capabilities.
cold start, 51-52
reload, 59-60
system generation, 46-47

Batch Processing, The execution of programs one after another, in the order in which they are entered into the 1800, done in a system that is not capable of responding to real-time events.

MPX Sample Program, 141

BDCRL
operating procedures, 77-79
punching, 31
system generation, 36-37

BDPAT
operating procedures, 80-82
punching, 31

BDPIP
operating procedures, 82-84
punching, 31
system generation, 32,37-38

BDUMP
operating procedures, 79-80
punching, 31
BDUPL  
operating procedures, 84-85  
punching, 31  
system generation, 32,37-38

BDWAP  
operating procedures, 86-87  
punching, 31  
system generation, 31,32-34

Binary Format 80

BLIST  
operating procedures, 88  
punching, 31

BOM  
See Basic Operating Monitor.

BRELD  
operating procedures, 88-89  
punching, 31

BSCIO, The input/output control subroutine for the communications adapter.  
assembly, 28-29, 40.2, 42.7  
equate cards, 28, 40.2, 42.6

BULKN, The input/output control subroutine for the 1810 disk storage unit and for mapped 1810 drives.  
disk dump, 93-94

Card Data Format (CDD) 138-140. The format in which information is punched into cards by a dump operation.  
output from BDCRL, 78-79, 139  
output from BDUMP, 138-140

Card Dump Analysis Program (DMPAN) 104-111

Card Dump Program, 1442 Relocatable (CRDMP) 102-104

Card Read Punch, 1442  
See 1442 Card Read Punch.

Card Set  
BDPAT, 80-82

Card System Format (CDS) 135-137. The format in which an object program (the result of a compilation or an assembly) is punched on binary cards.

Card Utilities, BOM 77-79  
See also BDCRL, BDPAT, BDPIP, BDUMP, BDUPL, BDWAP, BLIST, BRELD.

Cartridge, 2315 Disk  
See 2315 Disk Cartridge.

CDD Format  
See Card Data Format.

CDS Format  
See Card System Format.

CECLD 115-118, 121  
See also CE Coreload.

CECLX 115-123  
See also CE Coreload.

CE Core, A block of 512 words added to each 8K words of main storage, used by the system to log errors in the

Error Statistics Table, Trace Table, and CE Error Log.

CE Coreload (CECLD, CECLX) 115-123. The program that is used to print out and modify the status of input/output devices on the system.  
functions, 116  
system generation, 47,49  
1442 diagnostic, 125

CE Error Log 115,119,120. An area of CE Core where information about various errors is recorded as the errors occur.

Central Processing Unit (CPU). The unit of the 1800 that contains circuits that control the interpretation and execution of instructions.

Check/Stop Trace 100-101  
table dump, 97

Clearing Main Storage 15

Clock  
setting, 56,60

Cold Start 51-58. The process of starting operation of the MPX system, using the Cold Start Program.  
See also Cold Start Coreload, Cold Start Loader, Cold Start Name Card, Cold Start Program, batch-processing system, 51  
real-time system, 51  
system generation, 17,18,48  
1442 diagnostic, 125

Cold Start Coreload 52,55. The coreload that is given control by the Cold Start Program.

Cold Start Loader 18,52  
cards, 53-54  
system generation, 27,47

Cold Start Name Card  
format, 52,55  
reload, 55,61  
system generation, 27,47

Cold Start Program, The program that loads the Executive or BOM into main storage, and gives control to a specified program.  
operation, 51,56-57  
system generation, 17

Communications Adapter (CA), An adapter that permits the 1800 to communicate with other computers and terminals by providing for the attachment of one or two line adapters.

CE Coreload extension, 119-120  
cold start name card, 55  
system generation, 18,28-29,40.1-40.2, 42.3-42.5

Compiler, FORTRAN  
deleting from disk, 73-74

Completion Code  
BULKN, 94

Compressed Binary Format 103-104

Console Operations  
alter main storage, 16
clear main storage, 15
console program load, 15
display main storage, 16
operations monitor, 16
sense, program and data switch operations, 150.1
Console Program Load 15
Control Unit, 2841
See 2841 Storage Control.
Copying Disks
See BDUPL.
Coreload, An executable absolute program or program portion (link), stored on disk and loaded into a partition for execution.
building, 17, 47-48
CE, 115-124
rebuilding
conditions 50.1
example 50.2
Coreload Area
See Partition, VCORE.
Coreload Map 112
Core Storage
See Main Storage.
CRDMP 102-104
CSPAR
assembly, 40, 42.2
Cylinder, On a 2315 disk cartridge, a track on the upper surface of the disk together with the corresponding track on the lower surface. On a 1316 disk pack, a set of ten parallel tracks, one on each recording surface.
Cylinder Break Card 78-79, 80, 138
Cylinder Data Card 78, 80, 139
Data Channel, An interface used to connect I/O devices to the CPU.
Data Communication System
See 2790 System.
Data, Console Sense, and Program Switch Operations, 150.1
Data Formats
See Binary Format,
Card Data Format,
Card System Format,
Compressed Binary Format,
Disk Data Format,
Disk System Format,
Print Data Format.
Date
entering, 56, 60
DDF
See Disk Data Format.
Debugging Aids 91-125
Decimal/Hexadecimal Disk Addresses 131
DEFINE NDISK 45
Deleting System Programs from Disk 73-74
Diagnostic, 1442 Card Read Punch 124-125
Disk Address Card 81-82
Disk Addresses, decimal and hexadecimal 131
Disk Cartridge, 2315
See 2315 Disk Cartridge.
Disk Data Format (DDF) 135. The format in which information is stored by the Disk Management Program function *STOREDATA.
Disk Dump
See Dump.
Disk Management Program (DMP), A group of disk utility and maintenance programs that operate under control of the Batch-Processing Monitor Supervisor.
input to, 64-65
Disk Pack, 1316
See 1316 Disk Pack.
Disk Pack Initialization Program (DPIP) 101
Disk Storage Drive, 2311
See 2311 Disk Storage Drive.
Disk Storage Unit, 1810
See 1810 Disk Storage Unit.
Disk System Format (DSF) 133-135. The format in which an object program (the result of a compilation or an assembly) is stored on disk. Programs stored by Disk Management Program functions other than *STOREDATA are stored in Disk System Format.
Displaying Main Storage 16
DMON 67-76
calling sequence, 67
deleting system programs, 73-74
functions, 67
installing PTFs, 72
loading, 127
loading new versions of programs, 74-76
patching disk words, 69-70
patching system programs, 71
printing PTF log, 72-73
replacing, 76
replacing subroutines, 75-76
updating version/modification, 73
DMP
See Disk Management Program.
DMPAN 104-111
DMPS
assembly, 40, 42.2
equate cards, 36, 39
DPIP (Disk Pack Initialization Program) 101
DSF
See Disk System Format.
DTRAP 111-114
Dump, To copy data from storage to an output device or to another part of storage; also, the copy so obtained.
disks to cards during system generation, 28, 36-37
disks to printer, 92-96, 101-102
main storage to cards, 18, 102-104
main storage to printer, 96-97, 101-102
table, 97
1316 disk pack to cards, 77-78
1810 drive to cards, 79-80
Duplicating Disks
See BDUPL.
FORTRAN, FORMula TRANslating system; a procedure-oriented programming language.

Logical Unit Number, 127-129

FORTRAN Compiler, The program that generates a machine-language program from a FORTRAN-written program.

deleting from disk, 73-74

Full Trace 98-99

Header Card, ISS Subroutine 137

Header Card, Mainline 135-136

Header Card, Program 133-134

Header Card, Subroutine 137

Hexadecimal/Decimal Disk Addresses 131

High Core Loader 18-20

IAC

See Interrupt Assignment Code.

Initialization

1316 disk pack, 82-84, 101

2315 disk cartridge, 86-87

Input/Output Devices 1-14

See also specific devices, such as 1442 card read punch.

Interrupt, The recognition by the 1800 of an event that alters the sequence of program execution by causing execution of a specific program.

Interrupt Assignment Code (IAC) 127-129

Interrupt Coreload Table Map 50.6

Interrupt Level, One of up to 24 categories to which interrupts can be assigned to specify their relative importance in being recognized and serviced.

assignment, 17

Interrupt Servicing, The execution of a sequence of instructions in response to an interrupt.

I/O (input/output)

ISS Header Card 137

K, A symbol for the number 1024.

Keyboard, of 1816 Printer Keyboard See also 1816 Printer Keyboard.

input to Supervisor, 64-65

Labeling Disks 45

LET

See Location Equivalence Table.

Library, Subroutine

See Subroutine Library.

List Printer, The printer that lists all control statements being processed, all informational messages pertaining to user programs, and optionally, all statements being processed by the language translators.

DMON, 70
dumps, 92-96, 101-102

system generation, 18

Listing Cards 88, 102
Loader
See Absolute Loader,
High Core Loader,
System Loader.

Loading New Versions of System Programs 74-76

Location Equivalence Table (LET), The table that contains information about the contents of the System Area, the User Area, and Batch-Processing Working Storage of a particular 1810 drive.

Logical Unit Assignments changing, 117
Logical Unit Number (LUN) 127-129
LSPCL assembly, 40,42.3
equate cards, 36,39
LUN
See Logical Unit Number.

Macro Assembler, The translating program that accepts as input assembler-language instructions, pseudo-operations, and macro instructions.
DOM assembly, 39
deleting from disk, 73-74
Executive Director assembly, 39,42.1
output listing, 65-66.2
Subroutine Library assembly, 39

Magnetic Tape Unit, 2401/2402
See 2401/2402 Magnetic Tape Unit.

Mainline Header Card 135-136

Main Storage
altering, 16
clearing, 15
defining layout, 45-46
displaying, 16
dumping, 96-97

Mapped 1810 Drive, A data set on a 1316 disk pack that functionally appears to the MPX system as the contents of a 2315 disk cartridge.
cold start, 52,56
duplicating, 85

Master Interrupt Control (MIC), The program that passes control to the appropriate interrupt-serving program whenever an external, input/output, or programmed interrupt occurs.

MDFIO
assembly, 40.1,42.3
equate cards, 36,39

MPFO
assembly, 40.1,42.3
equate cards, 36,39

MLIST 102
Modification Number
updating, 73

Monitor Trace 99-100

MOPTP 110.1-110.2

MPX Sample Program
See Sample Program, MPX.

Multiprogramming, A technique for executing numerous programs simultaneously in a single CPU by means of an interweaving process.

Multiprogramming Executive Operating System (MPX), An operating system for the 1500 that can control processes and provide multiprogramming and background processing.

Name Card, Cold Start format, 52,55
reload, 55,61
system generation, 27,47

NDISK 45

Object Code, Machine-language instructions that are output from a language translator.

Offline Device, A device that is not under control of the operating system.

OLDMP (Online Dump Program) 101-102

Online Device, A device that is under control of the operating system.

Online Dump Program (OLDMP) 101

Operations Monitor 16

Optional Material Tape Dump Program (MOPTP) 110.1-110.2

Output Listing, Macro Assembler 65-66.2

Pack, 1316 Disk
See 1316 Disk Pack.

Paper Tape
See 1054 Paper Tape Reader,
1055 Paper Tape Punch.

Paper Tape Punch, 1055
See 1055 Paper Tape Punch.

Paper Tape Reader, 1054
See 1054 Paper Tape Reader.

Parity Error
card read operation, 5

defining, 45-46

Patch Record
*MON, 70
BDPAT, 81-82

Patching Disk Words
*MON statement, 69-70
BDPAT, 80-82

binary patch card, 70

hexadecimal patch record, 70

Patching System Programs 71

PID
See Program Information Department.

Plotter, 1627
See 1627 Plotter.

PRD Format
See Print Data Format.

Print Data Format(PRD) 140. The format in which programs and data are dumped to printers and typewriters.
Printer, 1053
See 1053 Printer.
Printer, 1443
See 1443 Printer.
Printer Keyboard, 1816
See 1816 Printer Keyboard.
Priority
1. A number assigned to a coreload queued to be executed on a level. It specifies precedence of the coreload within the queue.
2. A parameter of a call to an input/output control subroutine, used to specify the precedence of the call within the queue of requests for use of the input/output device.
Process, A device or set of devices monitored or controlled by a processor-controller.
Process Coreload, A coreload that is executed in response to a real-time event.
Processor-Controller, The unit that contains the central processing unit, main storage, the circuitry and controls necessary for attachment of process I/O, and the logic necessary to provide real-time system capabilities.
Program Header 133-134. A record that precedes a program stored on disk in disk system format; it contains various indicators about the program.
Program Information Department (PID) 17,18
Program Temporary Fix (PTF)
See also PTF Log.
installing, 67-68,72
Program, Console Sense, and Data Switch Operations 150.1
PTF
See Program Temporary Fix.
PTF Log
printing, 68,72-73
Punch, Card
See 1442 Card Read Punch.
Punch, Paper Tape
See 1055 Paper Tape Punch.
Queue, A waiting list for the use of some system resource, such as the CPU or an I/O device.
Queueable Coreload, A coreload that, as a result of an interrupt, can be queued for execution on a level.
Reader, Card
See 1442 Card Read Punch.
Reader, Paper Tape
See 1054 Paper Tape Reader.
Real-Time System, A system in which computation is carried out during or immediately following the actual time in which the related physical process takes place, so that the results of the computation may be used in guiding the physical process. The time element involved is generally considered to be in the subsecond range.
cold start, 51
reload, 59,61
system generation, 47-48
Rebuilding
Executive
conditions 50
example 50.2
Coreload conditions 50.1
Reload 59-61
BOM, 59-60,91-92
Executive, 61,91-92
1810 drive, 88-89
2311 drive, 78-79
Reload Coreload 55,61
Relocatable Card Dump Program, 1442
(CRDMP) 102-104
Relocatable Program, A program that has been assembled or compiled, may be stored in the User Area in disk system format, and can be executed in different places in main storage. Its instructions do not contain actual addresses; it must be built into a coreload before it can be executed.
Replacing DMON 76
Replacing System Programs on Disk 74-76
Retry Counter, A counter that keeps track of how many times an erroneous I/O operation has been retried.
Sample Program, MPX 141-150
punching, 31
Sector, On a 2315 disk cartridge, one-quarter of a track; 320 words.
Sector Break Card 137
Sequence Numbers
checking, 88,102
Serial Number, Volume
checking, 83
SET 64
Source Code, Code that is input to a language translator.
SPAR Coreload, A coreload that, upon being placed in main storage, remains there until the user takes specific action to make the partition available for another use. It is used as an extension of the Executive.
reload, 61
Storage Control, 2841
See 2841 Storage Control.
Storage Protection, An arrangement to prevent writing in certain areas of main storage.
card read operation, 5
Subroutine Header Card 137
Subroutine Library

DPIE, 101
equate cards, 17, 30, 35-36, 38-39
MLIST, 102
OLDMP, 101-102
replacing subroutines, 75-76
SUP
See Batch-Processing Monitor
Supervisor.
System Generation, The process during
which the capabilities, contents,
and organization of a particular MPX
system are defined and established.
with CA 28, 17-50
introduction, 17
phase 1, 17, 22-23
phase 2, 17, 23-43
phase 3, 17, 44
phase 4, 17, 45-46
phase 5, 17, 46-50
one 1810 drive, 21, 23-29, 44-50
one 2311 drive, 21-23, 29-32, 36-37,
38-42, 14, 44-50
two or more 1810 drives, 21-23,
29-36, 42, 1-50
two or more 2311 drives, 21-23,
29-32, 37-42, 1, 44-50
System Generation Monitor
functions, 17
System Loader
assignment cards, 17, 44
loading DMON, 76
Punching, 30-31
System Printer, The printer that lists
all Supervisor, Disk Management
Program, and Builder control
statements, all informational
messages pertaining to system
programs, and all error messages
except those issued by Error Alert
Control,
system generation, 18, 29
trap subroutines, 111
System Residence Disk, A disk pack
or cartridge containing copies of all
programs defined for an MPX system,
together with storage areas and
tables.
Table Dump 97
Tape, Magnetic
See 2401/2402 Magnetic Tape Unit.
Tape, Paper
See 1054 Paper Tape Reader,
1055 Paper Tape Punch.
Time
entering, 56, 60
Timeout, The time interval allowed for
something to happen before a timer
generates an interrupt.
Timer, A clocking device that generates
an interrupt each time a specified
time interval elapses.
cold start, 51
Trace, communications Line 119
Trace Define Limit Subroutine 98-101
Trace Subroutines 97-101
Track, A circular path on a disk along
which data can be recorded; 1280
words on a 2315 disk cartridge,
3625 bytes on a 1316 disk pack.
TRAP 111-114
Utilities 77-89
See also BDCRL, BDPAT, BDPIP,
BDUMP, BDUPL, BDWAP, BLIST,
BRELD.
punching, 31
Utility Monitor 91-97
User Library
See also Subroutine Library.
CE Coreload, 115
Variable Core
See VCORE.
VCORE, The partition of main storage
in which background-processing
programs, batch-processing pro-
grams, and interrupt coreloads
must be executed, and queueable
coreloads can be executed.
cold start coreload, 61
disk dump, 94
DMPAN dump, 104
disk dump, 107
DMPAN execution, 107
Version Number
updating, 73
Volume Serial Number
checking, 83
Volume Table of Contents (VTOC)
constructing, 82
dumping, 78
writing, 82
Word, The 16-bit basic unit of storage in
the 1800 system.
XREF Table, Macro Assembler 66
1053 Printer
cold start, 57-58
loading procedure, 6
operation, 5-6
readying procedure, 6
1054 Paper Tape Reader
operation, 6
readying procedure, 6
1055 Paper Tape Punch
loading procedure, 7
operation, 7
readying procedure, 7
1316 Disk Pack
See also Dump, Reload.
duplication, 84-85
initialization, 82-84,101
system generation, 18,32,36-38
1442 Card Read Punch
diagnostic, 124-125
errors and recovery procedures, 1-5
input to Supervisor, 64
operation, 1-5
readying procedures, 1
1442 Relocatable Card Dump Program (CRDMP) 102-104
1443 Printer
loading procedure, 12
operation, 11-13
readying procedure, 11
system generation, 29
1627 Plotter
loading procedure, 9
operation, 9-10
readying procedure, 9-10
1810 Disk Storage Unit
See also Dump, Patching Disk Words, Reload.
cold start, 56-57
duplicating drives, 84-85
loading procedure, 8-9
operation, 8-9
readying procedure, 8
system generation, 18,21-36,42.1-50
1816 Printer Keyboard
input to Supervisor, 64-65
loading procedure, 6
operation, 5-6
readying procedure, 6
2311 Disk Storage Drive
See also Dump, Reload.
CE Coreload, 120-120.1
cold start, 56-57
duplicating drives, 84-85
loading procedure, 14
operation, 13-14
readying procedure, 14
system generation, 18,21-23,29-32, 36-42.1,44-50
unloading procedure, 14
2315 Disk Cartridge
duplicating, 32,37-38,84,85
initializing, 86-87
labeling, 45
system generation, 18,32
2401/2402 Magnetic Tape Unit
loading procedure, 11
operation, 10-11
readying procedure, 10-11
unloading procedure, 11
2790 System
CE Coreload, 120.1-120.5
Error Sort Program, 120.2-120.3
I/O Devices (See the 2790 Data Communication System, Component Description, Order Number GA27-305.)
Exerciser/Diagnostic Program, 42.1, 43
system generation, 41-42, 42.4-42.6, 42.8-43
2841 Storage Control
address, 55,56
cold start name card, 55
starting procedure, 14

158 1800 MPX Operating Procedures
IBM 1800 MPX Operating System
Operating Procedures
GC26-3725-4

Your views about this publication may help improve its usefulness; this form will be sent to the author's department for appropriate action. Using this form to request system assistance or additional publications will delay response, however. For more direct handling of such requests, please contact your IBM representative or the IBM Branch Office serving your locality.

Possible topics for comment are:
Clarity  Accuracy  Completeness  Organization  Index  Figures  Examples  Legibility

What is your occupation? ____________________________

Number of latest Technical Newsletter (if any) concerning this publication: ____________________

Please indicate your name and address in the space below if you wish a reply. ____________________________________________

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments.)
Your comments, please...

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. Your comments on the other side of this form will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Business Reply Mail
No postage stamp necessary if mailed in the U.S.A.

IBM Corporation
Systems Publications, Dept 27T
P.O. Box 1328
Boca Raton, Florida 33432

International Business Machines Corporation
General Systems Division
5775D Glenridge Drive, Atlanta, Georgia 30328
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)