60456530

CONTROL DATA

MSL 15X
OFF-LINE MAINTENANCE
SOFTWARE LIBRARY
REFERENCE MANUAL

CDC® COMPUTER SYSTEM:

CYBER 170 MODELS 815, 825, 835
845, AND 855
CYBER 180 MODELS 810, 830, 835, 840,
845, 850, 855, 860, 960, 962, 990, 990E,
992, 994, AND 995E

Manual History

Revision Level	System Version	Date
A	Manual released on microfiche.	11-04-80
B	Manual revised to release R01 L124. Revision includes an update to CTI/HVS. Miscellaneous corrections are also made.	04-15-81
C	Manual revised to include procedure for CYBER 170 Model 825. Miscellaneous corrections are also made.	09-02-81
D	Manual revised to release L132. Revisions include documentation of CAU utility and enhancements to HDP, CDT and TDX. Revision includes addition of the former DEMOT reference manual in section 5. Due to extensive changes, change bars and dots are not used and each page reflects the current revision.	12-18-81
E	Manual revised to release L137. Manual includes procedures for CYBER 170 Model 855 Computer System. Revision includes addition of VP, VC, and VK commands, on-line edit utility, and miscellaneous corrections. This revision is a printed manual that obsoletes all previous microfiche editions of this manual.	06-18-82
F	Manual revised to release L143. Manual includes procedures for CYBER 170 model 815 Computer System. Revision includes addition of power-on initialization procedures and miscellaneous corrections. This edition obsoletes all previous editions.	11-22-82
G	Manual revised to release L149. Manual includes a description of automatic PP assignment, revised CMSE PP displays, MSL 15X name modification, and miscellaneous corrections.	05-02-83
H	Manual revised to MSL release L156; CIP release L001. Manual includes procedures for CYBER 170, Models 810, 830, and 845 Computer Systems. Revision includes support of the 834 (ISD) Disk Subsystem, revised installation procedures, and miscellaneous corrections. The revision also deletes descriptions of CTI displays from section 2. This edition obsoletes all previous editions.	11-18-83

J	Manual revised to MSL release L161; CIP release L002. Manual includes support of C180 Models 810, 830, 835, 845, and 855, support of 721 and 752-Compatible Consoles, support of 639 intelligent small tape (ISMT) unit, and miscellaneous corrections. This revision obsoletes all previous editions of this manual.	05-15-84
K	Manual revised to MSL release L167; CIP release L003. Changes include support of models 840, 850, and 860; support of dual CPU and shadow memory; revisions to control store and maintenance channel commands; revisions to DEMOT displays; and miscellaneous corrections. This revision also deletes documentation of the EDIT On-Line Utility.	12-03-84
L	Manual revised to MSL release L173; CIP release L004. Changes include addition of Disk Format Utility (DFU) and deletion of Remote Terminal Driver (RTD). (LR command replaces RTD.)	05-03-85
M	Manual revised to MSL release L179; CIP release L005. Changes include addition of command buffer editing utility (CBU), CMSE area of CM, support of C180 Model 990, CTI support I4, CMSE disk driver changes, and miscellaneous changes, additions, and corrections.	11-22-85
N	Manual revised to MSL release L186; CIP release L006. Changes include addition of capture buffer dump utility (CBD), I4 mode support of I4 IOU, and miscellaneous corrections.	05-13-86
P	Manual revised to MSL release L678; CIP release L007. Changes include, replacement of VLEX routine with VEXC and miscellaneous corrections.	04-21-87
R	Manual revised to MSL release L688; CIP release CV008. Changes include expanding Display Virtual Memory (AV, BV) and miscellaneous changes.	09-21-87
T	Manual revised to MSL release L704; CIP release CV009. Changes include support of dual I4 IOU, single I4C IOU, CYBER 960, and miscellaneous changes.	06-01-88
U	Manual revised to MSL release L710; CIP release CV009. Changes include support of dedicated load device.	09-01-88
V	Manual revised to MSL release L727; CIP release CV010. Changes include reorganization of material and miscellaneous corrections. All references to IOUs as I1, I2, I4, and I4C have been replaced by actual IOU equipment numbers. This revision obsoletes all previous editions of this manual.	05-01-89

W	Manual revised to MSL release L739; CIP release CV011. Changes include deletion of material which exists in the CIP reference manuals, along with other miscellaneous updates.	11-17-89

Revision W of this manual updates documentation to CIP L739. Due to the number of pages affected by changes for this release, this edition obsoletes all previous editions.

Contents

Figures

Tables

About This Manual

Organization

The CONTROL DATA® Maintenance Software Library (MSL) 15X described in this reference manual consists of a set of tests and diagnostics that aid in the checkout and isolation of defective components in the CDC® CYBER 170 Models 815, 825, 835, 845, and 855 Computer Systems; CYBER 180 Models 810, 830, 835, 840, 845, 850, 855, 860, and 990 Computer Systems; CYBER 960, 962, 990E, 992, 994, and 995E Computer Systems.

The MSL includes the common maintenance software executive (CMSE), which controls the execution of diagnostics, tests, and utilities, as directed by the operator.

Audience

This manual contains a description of MSL for use by Control Data customer engineers.

Model Classification by IOU Configuration and Upgrade

Certain procedures within this manual pertain only to computer systems of a common input/output (IOU) type, where the specific IOU type is referenced by its equipment number. For instance, a procedure which is only applicable to systems with an AT511/AT512 IOU would pertain only to CYBER 962/992 systems. Refer to the following table.

Primary IOU Equipment Number	Applicable Computer Systems
AT478/AT481	CYBER 840A, 850A, 860A, 870A, 960, and 994. CYBER models 990, 990E, and 995E. CYBER *IOU-upgraded* models 840, 845, 850, 855, and 860.
AT511/AT512	CYBER 962 and 992.

Conventions

The following conventions are used in this manual:

- When the word "press" is used in an instruction, it tells you to press the named key or keys. For example, when you see the instruction:

  Press **(CR)**

  you should press the carriage return key on a CC545 or CC634B console keyboard.

  If you are using a CC598A console and see the instruction:

  Press **Enter**

  you should press the key labeled **Enter** on your console keyboard.

  (Although the space bar is not labeled, bold face is used to name that key also, for example: press **Space Bar**.)

- When the word "press" precedes a string of keys that are separated by hyphens, it means that you should simultaneously press the keys separated by hyphens. For example, when you see the instruction:

  Press **Ctrl-F2**

  you should simultaneously press the keys marked **Ctrl** and **F2** on your console keyboard.

- When the word "type" is used in an instruction, it tells you to type in the following word.

  For example, when you see the instruction:

  Type TEST

  you should type the letters T E S T in sequence.

This manual includes many procedures which must be executed from your primary system console. Throughout this manual you will be directed to press the DEADSTART button on your system console to initiate the deadstart process.

Related Publications

Descriptions of the tests and diagnostics are provided in the following related publications.

Manual Title	Publication Number
MSL 15X Model Independent Maintenance Software Reference Manual	60469390
MSL 151 CYBER 170 Models 815 and 825, CYBER 180 Models 810 and 830 Maintenance Software Reference Manual	60469400
MSL 152 CYBER 170/180 Model 835 Maintenance Software Reference Manual	60469410
MSL 153 CYBER 170/180 Models 840, 845, 850, 855, and 860 Test Procedures Maintenance Software Reference Manual	60459140
IOU2 - MSL 153 Test Description	60460000

Manual Title	Publication Number
ACT3/P - MSL 153 Test Description	60460010
ANT3/P - MSL 153 Test Description	60460020
BPT3/P - MSL 153 Test Description	60460030
CMT3/P - MSL 153 Test Description	60460040
CST3/P - MSL 153 Test Description	60460050
CTT3/P - MSL 153 Test Description	60460060
ICT3/P - MSL 153 Test Description	60460070
IFT3/P - MSL 153 Test Description	60460080
LMT3/P - MSL 153 Test Description	60460090
MAT3/P - MSL 153 Test Description	60460150
MCT3/P - MSL 153 Test Description	60460100
MDT3/P - MSL 153 Test Description	60460110
OIT3/P - MSL 153 Test Description	60460120
PDT3/P - MSL 153 Test Description	60460130
SMT3/P - MSL 153 Test Description	60460140
MSL 153/155 CYBER 96X Maintenance Software Reference Manual Test Procedures	60461110
MSL 153/155, CYBER 96X Maintenance Software Reference Manual Test Descriptions	60461920
ACT5 - MSL 155 Test Description	60461120
BPT5 - MSL 155 Test Description	60461130
CMT5 - MSL 155 Test Description	60461140
EPT5 - MSL 155 Test Description	60461150
FPT5 - MSL 155 Test Description	60461160
IDT5 - MSL 155 Test Description	60461170
IFT5 - MSL 155 Test Description	60461180
IGT5 - MSL 155 Test Description	60461190
LST5 - MSL 155 Test Description	60461200
MAT5 - MSL 155 Test Description	60461210
MIT5 - MSL 155 Test Description	60461220
MST5 - MSL 155 Test Description	60461270

Manual Title	Publication Number
OCT5 - MSL 155 Test Description	60461230
PAT5 - MSL 155 Test Description	60461240
SCT5 - MSL 155 Test Description	60461250
VAT5 - MSL 155 Test Description	60461260

The following documents contain information pertaining to the MSL library.

Manual Title	Publication Number
Assembly Buffer Utility Reference Manual	60469330
CML Reference Manual	60455980
CYBER Initialization Package (CIP) CYBER 180 Models 810, 815, 825, 830; CYBER 810A, 830A Computer Systems Reference Manual	60000417
CYBER Initialization Package (CIP) CYBER 180 Models 835, 845, 855; CYBER 840, 850, 860 Computer Systems With IOU AB115A Reference Manual	60000418
CYBER Initialization Package (CIP) CYBER 180 Models 845, 855; CYBER 840, 850, 860 With IOU AT478A/AT481A; CYBER 840A, 850A, 860A, 870A, 990, 990E, 995E Computer Systems Reference Manual	60000419
CYBER Initialization Package (CIP) CYBER 960, 994 Computer Systems Reference Manual	60000420
CYBER Initialization Package (CIP) CYBER 962, 992 Computer Systems Reference Manual	60000421
CYBER Initialization Package (CIP) CYBER 170 Models 865, 875; Non-Model 8XX/9XX Computer Systems Reference Manual	60000422
CYBER 170/180 Models 835, 840, 845, 850, 855, 860, and 990, CYBER 990E, 995E, and 994 (CYBER 170 State) Hardware Reference Manual	60469290
CYBER 180 Model 990, CYBER 990E, 995E, 992 and 994 (Virtual State) Hardware Reference Manual, Volume 1	60462090
CYBER 170/180 Models 810, 815, 825, 830, 835, 840, 845, 850, 855, 860, and 990, CYBER 990E, 995E, 992, and 994 (Virtual State) Hardware Reference Manual, Volume 2	60458890

Manual Title	Publication Number
CYBER 170 Models 815 and 825 (CYBER 170 State) Hardware Reference Manual	60469350
CYBER 180 Models 810 and 830 (CYBER 170 State) Hardware Reference Manual	60469420
Intelligent Small Magnetic Tape (ISMT) Subsystem Reference Manual	60461090
MALET Reference Manual	60456020
MSL 100 Off-Line Maintenance System Library Reference Manual	60455770
MSL 100/140 Instant Manual	60456900
MSL 15X CMSE Command Card	60456600
CDC 721-21/31 Owner's Manual	62950101
CDC 19003 (CC598-A/B) System Console Operations and Maintenance Guide	60463610
CYBER Systems Peripheral Diagnostic Reference Manual	60000144
CDC CYBER 96X Maintenance Software Library Reference Manual	60000292

Disclaimer

This product is intended for use only as described in this document. Control Data cannot be responsible for the proper functioning of undescribed features or parameters.

Introduction 1

Introduction 1

The Maintenance Software Library (MSL) 15X is a set of tests, diagnostics, and utilities that isolates malfunctions, tests system components, and monitors machine states. The MSL is divided into the following parts.

- Common test and initialization (CTI) programs
- Common maintenance software executive (CMSE) programs
- Interface routines
- Hardware initialization and verification software (HIVS)
- Tests and diagnostics
- Utility routines
- Command buffers

CYBER Initialization Package (CIP)

Control Data creates a unique release tape, the CYBER Initialization Package (CIP), for each model of computer system. Microcode, environment interface (EI), the system console driver (SCD), the monitor display driver (MDD), CTI, CMSE, and selected MSL programs, command buffers, and utility routines are distributed on the CIP tape to sites with a maintenance contract. Sites without a maintenance contract receive a CIP tape that contains microcode, EI, CTI, CMSE, and a subset of MSL tests and utilities called HIVS.

CTI

CTI performs hardware intialization and represents a standardized human interface to the deadstart process. CTI consists of a set of displays from which you select the desired method for deadstarting. The CTI process includes some input/output unit (IOU) testing and hardware initialization. The initial displays and options offered by CTI are described in the appropriate CIP reference manual.

CMSE

CMSE provides an off-line monitor capability. Monitoring can be done from either a local or remote console. CMSE also provides display facilities, a keyboard command structure, a loading capability, and diagnostic sequencing. CMSE may be initialized following a short or long deadstart and an extended deadstart.

Interface Routines

The MSL contains routines that interface between tests and utility programs. Three routines (VLEX, EXC, and DEMOT) are described in this manual because they contain portions (parameters, error messages, and so on) that are visible to you.

Tests and Diagnostics

The MSL 15X library contains a set of tests for correct operation as well as diagnostics that isolate system faults. The MSL library is used as a source library for creating CIP, specifically for each model or models of computer systems. Refer to the table below.

MSL Library Name	System Models	CIP Tape Name
MSL151	810/815	CIP 810/815
	825/830	CIP 825/830
MSL152	835	CIP 835
MSL153	840	CIP 840
	845/850	CIP 845/850
	855/860/870	CIP 855/860/870
MSL155	990/995	CIP 990/995
	992	CIP 992
	994	CIP 994
CYBER 96X	960	CIP 960
	962	CIP 962

Appendix D of this manual provides a list of tests and diagnostics available on each CIP binary tape.

The term MSL 15X, as used in this manual, refers to the MSL 15X library and any of its subsets (unless otherwise noted). Descriptions of each test or diagnostic are not contained in this manual, but are contained in separate MSL fault detection and isolation maintenance software manuals as listed in About This Manual.

Hardware Initialization and Verification Software (HIVS)

The hardware initialization and verification software (HIVS), a basic subset of MSL tests, performs confidence testing of the system. The HIVS module exists on the HIVS CIP tape and is distributed to sites without a maintenance contract.

CTI initiates the hardware verification sequencer (HVS) which runs tests under control of CMSE.

Utility Routines

The utility routines provide features not available in standard tests. Two types of utilities exist in the MSL; those that are part of CMSE, which are usually overlays executed by a CMSE command, and standalone utilities, which may require a supporting system to perform their function.

Command Buffers

Command buffers are useful tools enabling you to load and execute a series of tests and to set parameters. Methods for constructing command buffers and examples of their use are described in section 4 of this manual.

NOTE

Release command buffers may not work on all configurations and may need to be modified.

A command buffer is a series of keyboard (CMSE and program) commands residing on the MSL disk that are executed as a group by CMSE. Each command buffer is assigned a unique name to identify it on the disk. Command buffers may be input interactively from a terminal. Refer to section 4 for a more detailed description.

Offline Maintenance 2

Offline Maintenance 2

This section describes offline maintenance options which must be performed from the INITIAL OPTIONS display shown in figure 2-1. Refer to the appropriate CIP reference manual for procedures on how to bring up this display.

Warmstart, coldstart, initialization, power-on initialization, long deadstart (LDS), and extended deadstart sequence (EDS) procedures are contained in the CIP reference manuals.

NOTE

For a deadstart from the CIP device (the deadstart device), the system shows the INITIAL OPTIONS display. If, in a dual IOU system, you deadstart from IOU1, the CMSE OFFLINE MAINTENANCE display shown in figure 2-3 would appear in place of the INITIAL OPTIONS display.

```
          INITIAL OPTIONS

A    OS LOAD AUTOMATIC
O    OS LOAD WITH INTERVENTION
U    UTILITIES
M    OFF-LINE MAINTENANCE
H    HELP

(CR) OS LOAD AUTOMATIC


CONTROL DATA PROPRIETARY PRODUCT
COPYRIGHT CONTROL DATA 1980

ppp-     VIII
XXXX
```

CIP01A

Figure 2-1. Initial Options Display, Deadstart From CIP Device

Typing M and pressing **(CR)** while the INITIAL OPTIONS display appears loads the CMSE test loader program.

The test loader presents the display shown in figure 2-2. The test loader responds to the following list of keyboard commands. Unless the command is a **(CR)** or *, all commands are terminated by a **(CR)**.

Command	Description
carriage return	Load CMSE from the designated library device.
right blank or +	Load CMSE from the designated library device. Increments the line number of the initial CMSE display by one. The parameter is cleared and the command is ready for parameter input. If right blank or + is entered repeatedly with no input, the line number is incremented without command execution.
mne	Load the standalone program named mne from the deadstart library device into PP 0 and initiate the program at address 0001 in the PP.
TDX	Load the tape-to-disk standalone routine (TDX) into PP0 to dump the MSL binary tape to an 834, 836, 844, 885, or 895 disk. All operating instructions for this dump are displayed by TDX. Refer to TDX description in section 6, Utility Programs.
KP	Clear all PP memories.
KC	Clear CM. The KC command fails when bringing up CMSE from tape after an initial power-on. This problem occurs on CYBER 170 models 815/825, CYBER 180 810/810A, 830/830A computer systems. After a power-on, the following CMSE commands must be executed before CM reads and writes can be performed. CX,M — Master Clear CMC CE,I — Clear IOU Errors CE,M — Clear Memory Errors ER,M,20,0 — Clear Memory Environmental Control Register ER,M,21,0 — Clear Memory Bounds Register If deadstart is from disk, CTI performs the equivalent of these commands before the system brings CMSE into execution. During an initial installation in which you deadstart from tape, the above commands should be executed.
*	Clear PP memories, CM and CE.
CE	Clear all error bits in maintenance register.
MR	Toggles MAINTENANCE REGISTER display. Attempts to turn off display if an error condition exists, but display is again presented because of the error condition. The error must be cleared before the display can be turned off.
nn.pp	Change the system configuration parameter (pp) for line number (nn) on the SYSTEM CONFIGURATION display entries.

```
            OPERATOR ENTRIES
            MNE(CR) LOAD STAND-ALONE PROGRAM
            N.YY(CR) CHANGE CONFIG. PARAM (N) TO (YY)
            (CR) LOAD MAINT. SYSTEM
            MR(CR) TOGGLE MAINT. REG. (MR) DISPLAY
            * KP+KC+CE            CE(CR) CLR MR
            KP(CR) CLR PP MEM     KC(CR) CLR CM


            1. SYSTEM TYPE                                    =  XXX
            2. CM SIZE                                        = XXXX
               ENTER SIZE IN MEGABYTES (DECIMAL)
            3. EM SIZE
               INVALID ENTRY FOR THIS SYSTEM TYPE
            4. NO. OF CPUS                                    =   XX
            5. NO. OF PPS (DECIMAL)                           =   XX
            6. MON PP/DIS PP                                  = XXXX
            7. PP COMM CH                                     =   XX
            8. D/S PP / USE CM                                =   02
            9. MSL CYL NO.                                    = XXXX

KYBD        ERROR                          MR ERROR      SS = XX


              D/S PP=(Deadstart PP) Flag bit 2**1.  CM=Usage
              flag bit 2**0 must be enabled for CIO PPs.
              B = This line is not to be displayed when the
              display is from the tape driver.
              KYBD = keyboard entry line (12 characters
              maximum).
              ERROR = error message area (20 characters
              maximum).
              MR ERROR = maintenance register error message
              (displayed only if a MR error is detected).
              SS = IOU status summary register contents.
```

Figure 2-2. CMSE Offline Maintenance Display (IOU0)

```
                          *** WARNING ***

         IF CENTRAL MEMORY IS TO BE USED FROM IOU1 FOR ANY
         REASON, COMMAND BUFFER EN2IOU MUST FIRST BE RUN FROM
         IOU0.


         ENTER A RETURN TO CONTINUE.




       OPERATOR ENTRIES
       MNE(CR) LOAD STAND-ALONE PROGRAM
       N.YY(CR) CHANGE CONFIG. PARAM (N) TO (YY)
       (CR) LOAD MAINT. SYSTEM
       MR(CR) TOGGLE MAINT. REG. (MR) DISPLAY
       * KP+KC+CE             CE(CR) CLR MR
       KP(CR) CLR PP MEM      KC(CR) CLR CM


       1. SYSTEM TYPE                                  =  XXX
       2. CM SIZE                                      = XXXX
          ENTER SIZE IN MEGABYTES (DECIMAL)
       3. EM SIZE
          INVALID ENTRY FOR THIS SYSTEM TYPE
       4. NO. OF CPUS                                  =   XX
       5. NO. OF PPS (DECIMAL)                         =   XX
       6. MON PP/DIS PP                                = XXXX
       7. PP COMM CH                                   =   XX
       8. D/S PP / USE CM                              =   02
       9. MSL CYL NO.                                  = XXXX

KYBD   ERROR                         MR ERROR        SS = XX


        D/S PP=(Deadstart PP) Flag bit 2**1.  CM=Usage
        flag bit 2**0 must be enabled for CIO PPs.
        B = This line is not to be displayed when the
        display is from the tape driver.
        KYBD = keyboard entry line (12 characters
        maximum).
        ERROR = error message area (20 characters
        maximum).
        MR ERROR = maintenance register error message
        (displayed only if a MR error is detected).
        SS = IOU status summary register contents.
```

Figure 2-3. CMSE Offline Maintenance Display (IOU1)

System Configuration Display Entries

Parameters in each line of the SYSTEM CONFIGURATION display shown in figure 2-2 can be changed by the following keyboard commands.

1. System Type

This command modifies the system type parameter and causes a corresponding change in the monitor name for CMSE loading.

Command format:

```
01.xxx or 1.xxx
```

where xxx is the 800 series model number (for example, xxx=810 or xxx=990).

2. CM Size

This command changes the amount of central memory that is available for CMSE use. The CM size must be a decimal count of the desired number of megabytes.

Command format:

```
02.xxxx or 2.xxxx
```

where xxxx is the number of megabytes of CM.

3. Extended Memory Size

This command is not valid for MSL 15X. It only appears to execute when entered.

4. Number of CPUs

This command changes the number of CPUs that CMSE recognizes as being in the system and available for use.

Command format:

```
04.xx or 4.xx
```

where xx is the number of CPUs in the system.

5. Number of PPs

This command changes the number of PPs available for use by CMSE. The number entered must be a decimal count of the PPs in the system.

Command format:

```
05.xx or 5.xx
```

where xx is the number of PPs in the system.

6. Monitor/Display Driver PP Numbers

This command changes the logical number of the PPs into which the monitor and display driver programs are loaded.

Command format:

```
06.xxyy or 6.xxyy
```

where xx is the monitor program PP number or I/O driver/monitor communication channel (if CM is not in use) and yy is the display driver program PP number or monitor/display driver communication channel (if CM is not in use).

7. PP Communication Channel Number

This command changes the number of the channel over which pool PPs and the CMSE monitor communicate.

Command format:

```
07.xx or 7.xx
```

where xx is the PP communication channel number.

8. Deadstart PP/CM Usage Flags

This command selects or deselects PP deadstart when CMSE is brought up and selects or deselects use of CM by CMSE.

Command format:

```
08.xx or 8.xx
```

where different values of xx result in the following:

xx=01 Use CM but do not initialize PPs.
xx=02 Initialize PPs but do not use CM (default parameter).
xx=03 Use CM and initialize PPs.

NOTE

For an AT478/AT481 IOU, option 01 or 03 must be selected in order for CMSE to control the CIO subsystem. If you select option 03, certain IOU diagnostics check CIO paths.

---

9. MSL Starting Cylinder

This command changes the starting cylinder for the MSL library. It allows access to more than one copy of MSL on the disk. When you enter the command followed by a (CR), the driver uses the cylinder entered as starting cylinder for bringing up CMSE. This command is invalid for tape driver CMSE loads.

Command format:

```
09.xxxx or 9.xxxx
```

where xxxx is the starting disk cylinder for the MSL library.

Figure 2-4 shows the initial CMSE OFFLINE DISPLAY for CYBER 962 and 992 computer systems.

```
OPERATOR ENTRIES
MNE(CR) LOAD STAND-ALONE PROGRAM
N.YY(CR) CHANGE CONFIG. PARAM (N) TO (YY)
(CR) LOAD MAINT. SYSTEM
MR(CR) TOGGLE MAINT. REG. (MR) DISPLAY
* KP+KC+CE              CE(CR) CLR MR
KP(CR) CLR PP MEM       KC(CR) CLR CM

1.  SYSTEM TYPE                                        =  962
2.  CM SIZE                                            = XXXX
    ENTER SIZE IN MEGABYTES (DECIMAL)                  =   XX
3.  NO. OF CPU'S                                       =   XX
4.  NO. OF PP'S (DECIMAL)                              = XXXX
5.  MON PP/DIS PP                                      =   02
6.  DP/CM


                  DEFAULT CONFIGURATION PARAMETERS USED

     KYBD               ERROR               MR ERROR            SS = XX

KYBD - keyboard entry line (12 characters maximum)
ERROR = error message area (20 characters maximum)
MR ERROR = maintenance register error message
           (displayed only if a MR error is detected)
SS = IOU status summary register contents
** = DP (Deadstart PP) Flag bit 2**1
```

Figure 2-4. CMSE Offline Maintenance Display (CYBER 962, 992 only)

CMSE Standalone Display Commands (CYBER 962, 992 Only)

CE - Clear Errors

This command is used to clear all errors in the IOU maintenance registers.

KP - Clear PP

This command is used to enter all PP memory locations with zeros except the first ten locations which may contain a PP idle package.

KC - Clear CM

This command is used to clear memory locations 0 through the last memory address as defined in the CM size parameter command.

MR - Toggle Maintenance Register Display

This command toggles the MAINTENANCE REGISTER display. If a maintenance register error condition exists, the command cannot turn off the display. The error must be cleared before the display can be turned off.

1. System Type

This command modifies the system type parameter causing a corresponding change in the monitor name for CMSE loading.

Command Format:

```
01,xxx or 1,xxx
```

where xxx is the CYBER 900 series model number.

2. CM Size

This command changes the amount of central memory that will be available for CMSE use. The CM size must be a decimal count of the desired number of megabytes.

Command Format:

```
02,xxxx or 2,xxxx
```

where xxxx is the number of megabytes of CM.

3. Number of CPUs

This command changes the number of CPUs available for use by CMSE.

Command Format:

```
03,xx or 3,xx
```

where xx is the number of CPUs in the system.

4. Number of PPs

This command changes the decimal number of PPs available for use by CMSE.

Command Format:

```
04,xx or 4,xx
```

where xx is the number of PPs in the system.

5. Monitor/Display Driver PP Numbers

This command defines which PPs are used for loading the CMSE monitor and display driver.

Command Format:

```
05,mmdd or 5,mmdd
```

where mm is the monitor PP number and dd is the display driver PP number.

6. Deadstart PP Flag

This command either selects or deselects deadstarting the PPs when CMSE is brought up and/or causes the CMSE monitor to load its overlays to CM.

Command Format:

```
06,xx or 6,xx
```

where different values of xx result in the following:

xx=01 CMSE monitor loads its overlay into CM.
xx=02 Initializes the PPs.
xx=03 CMSE monitor loads its overlay into CM and initializes the PPs.

CC634B Console Initialization Procedure

Before a CC634B console can be configured as either a primary or secondary console, you must first establish its operational state by installing a specific subset of its parameters. Although the following initialization procedure is time-consuming, once done you can perform subsequent initializations by simply pressing the console RESET button.

Parameters which require initialization can be found in the tables 2-1 and 2-2.

NOTE

This procedure assumes that the CC634B display terminal with no internal options installed is connected to a two-port multiplexer (TPM) and is operational.

Perform the following steps to initialize a CC634B console.

___ 1. Turn on the console. The MODE SELECTION display appears on the screen. This display consists of a row of 10 lighted blocks across the bottom of the screen. If the MODE SELECTION display does not appear, perform step 2; if it does, go to step 3.

___ 2. If the console has been previously configured to automatically select an operational mode, the MODE SELECTION display does not appear. In this situation, wait 60 seconds for a load timeout to occur. After this period of time the MODE SELECTION display should appear. If it does not, perform the following steps in order.

   ___ a. Press **SETUP**.

   ___ b. Press **F10** twice.

   The MODE SELECTION display should now appear.

___ 3. Press **CTRL-SETUP**. The default terminal installation parameters appear in a row of lighted blocks on the screen.

   NOTE

   A small blinking light appears in the F2 block. This light is a cursor. The cursor shows you where the next character you type on the keyboard will appear on the screen.

___ 4. Press **F4** to position the cursor under the F4 block (CONFIG).

___ 5. Type 1 to set auto select enabled.

___ 6. Press **F6** to position the cursor under the F6 block (AS X Y).

___ 7. Type 1 to select mode 1; CYBER mode.

___ 8. Press **COPY** to write the terminal installation parameters into nonvolatile memory. This makes the changes permanent. The cursor moves to the F9 block.

___ 9. Press **F10**, then type 1 to select operating mode 1 (CYBER mode) and display the Mode Installation Parameters.

___ 10. Press **F2**, if necessary, to position the cursor under the F2 block (CONFIG).

___ 11. Enter the value 1xxxx0 in the F2 block.

- ___ a. Type 1 to enable mode 1; CYBER mode.
- ___ b. Press the **Space Bar** until the cursor is under the sixth or rightmost position. Then type 0 to select host interface.

___ 12. Press **F3**, if necessary, to position the cursor under the F3 block (CONFIG).

___ 13. Enter the value xx0110 in the F3 block.

- ___ a. Space to the third position of the F3 block. Type 0 to select 7 data bits for the host.
- ___ b. With the cursor in the fourth position, type 1 to select parity enabled for the host.
- ___ c. With the cursor in the fifth position, type 1 to select even/mark parity for the host.
- ___ d. With the cursor in the sixth position, type 0 to select 1 stop bit for the host.

___ 14. Press **F4**, if necessary, to position the cursor under the F4 block (CONFIG).

___ 15. Enter the value 10xxxx in the F4 block.

- ___ a. With the cursor in the first position, type 1 to select data terminal ready (DTR) signal switched off.
- ___ b. With the cursor in the second position, type 0 to select request to send (RTS) signal on constantly when DTR or data set ready signals drop.

___ 16. Press **F5** to position the cursor under the F5 block (CONFIG).

___ 17. Enter the value 01xxxx in the F5 block.

- ___ a. With the cursor in the first position, type 0 to select pacing disabled.
- ___ b. With the cursor in the second position, type 1 to select bias enabled.

___ 18. Press **F6** to position the cursor under the F6 block (OPR DF). Four hexadecimal characters are displayed.

___ 19. Enter the value 0C05 in the F6 block.

___ 20. Press **F9** to position the cursor under the F9 block (DF T R).

___ 21. Press the **Space Bar** twice to position the cursor under the third hexadecimal character (under the T).

___ 22. Enter the proper transmit line speed/baud rate as follows:

Entry	**Baud Rate**
4	300 bps
5	600 bps
6	1200 bps
7	1800 bps
8	2400 bps
9	4800 bps
A	9600 bps
B	19,200 bps

___ 23. With the cursor under the fourth hexadecimal character (under the R), enter the proper receive line speed/baud rate using the listing in step 22.

The blocks that you changed should now be displayed at the bottom of the screen as follows:

F1	F2	F3	F4	F5	F6
RETURN	CONFIG 1xxxx0	CONFIG xx0110	CONFIG 10xxxx	CONFIG 010000	OPR DG 0C05

M03284

___ 24. Press **COPY** to write the Mode Installation Parameters into nonvolatile memory. This makes the changes permanent. The cursor moves to the F10 block.

___ 25. Press **F1** twice to return the console to CYBER mode.

Installation of parameters required to support automatic initialization of the CC634B console is now complete.

Table 2-1. Terminal Installation Parameters

F4	Configuration (binary value) Bit 1=1: Auto select enabled
F6	First character=1: CYBER mode selected

Table 2-2. CYBER Mode Installation Parameters

F2 Configuration (binary value)
Bit 1=1: Mode execution enabled
Bit 6=0: Host interface

F3 Configuration (binary value)

Bit 3=0: Host communications to have 7 data bits (excluding parity)
Bit 4=1: Parity in host communications enabled
Bit 5=1: Parity is even/mark
Bit 6=0: Words in host communications to have 1 stop bit

F4 Configuration (binary value)
Bit 1=0: DTR switched off during local operations
Bit 2=0: RTS constant

F5 Configuration (binary value)
Bit 1=0: Pacing disabled
Bit 2=1: Bias enabled

F6 Operation Default Parameters (hexadecimal value)

First hexadecimal character (binary power of two representation)
Bit 2^0=0: Online
Bit 2^1=0: Printer deselected

Second hexadecimal character (binary power of two representation)
Bit 2^3=1: Large CYBER

Third hexadecimal character (binary power of two representation)
Bit 2^0=0: Background dark
Bit 2^1=0: Cursor line
Bit 2^2=0: Cursor blink

Fourth hexadecimal character (binary power of two representation)
Bit 2^0=1: Full duplex
Bit 2^1=0: 80 characters per line
Bit 2^2=1: 30 lines
Transparent feature off

F9 Default File Number, Transmit/Receive Baud Rate (hexadecimal value)

Third character is the transmit baud rate as follows:

Value	Baud Rate
4	300 bps
5	600 bps
6	1200 bps
7	1800 bps
8	2400 bps
9	4800 bps
A	9600 bps
B	19200 bps

Fourth character is the receive baud rate. See transmit baud rate for values.

Primary and Secondary Console Use

Up to two display consoles can be configured simultaneously as operator consoles in CYBER 170 800 series and CYBER 180 computer systems.

Initially, however, because one console must be used to deadstart and initialize the system, this console is designated as the primary console until another deadstart is initiated or an LR command is entered designating a different console as the primary console.

The LR command, described in the CMSE Commands section of this manual, is used for assigning both primary and secondary consoles. Use of the LR command, in most cases, replaces the use of the remote terminal driver (RTD).

The following paragraphs define the hardware requirements and procedures governing use of primary and secondary display consoles.

Hardware Requirements

The CMSE display drivers support the following consoles.

- CC545 connected to channel 10_8.
- CC634B connected to a TPM on channel 15_8, set to a baud rate of 1200 to 19,200 bps.
- 752-compatible terminal connected to a TPM on channel 15_8, set to a baud rate of 300 to 19,200 bps.
- CC598-A/B connected to a TPM on channel 15_8, set to a baud rate of 300 to 19,200 bps.

If a CC545 is not in use on channel 10_8, CMSE supports a CC634B, CC598-A/B, or a 752-compatible terminal on the TPM as the primary console. If, however, a CC545 is in use on channel 10_8, CMSE will always support the CC545 as the primary console and support a CC634B, CC598-A/B, or a 752-compatible terminal on a TPM as a secondary console.

When both the primary and secondary consoles are in use, CC545 support is limited to displays that occur on the right screen. Additional information about keyboard and display use is described in the CMSE commands section of this manual.

When two consoles are in use simultaneously, the first character received from either keyboard causes the other keyboard to be locked out. In this case, CMSE does not accept characters from the second keyboard until it receives a **(CR)** from the first keyboard (or until an erase code is received from the second keyboard). Erase is used to unlock a keyboard if the operator at a console inadvertently enters a character.

Terminal Connection

The following procedure is used for bringing up and assigning a terminal as a primary or secondary console. For purposes of this description, this new console is initially referred to as the remote terminal.

At the remote terminal:

___ 1. Connect the terminal and apply power.

___ 2. Inform the user at the primary console of the need to be connected and the type of display being used.

At the primary console:

___ 1. Connect a telephone to a TPM.

___ 2. Enter the LR command with the appropriate parameters. See LR command description under CMSE commands.

At the remote terminal:

___ 1. Dial the appropriate telephone number to connect to a port of the TPM.

___ 2. Upon hearing a steady tone, switch the TALK/DATA switch on the modem to DATA and hang up.

___ 3. When the connection is successful, the CH/PP display appears. See Displays in the CMSE Commands section of this manual.

While CMSE is initializing the terminal, it displays the message CMSE on the terminal screen.

At the primary console:

___ 1. If the remote terminal fails to connect, the following message appears:

```
PORT x - NO DEVICE CONNECTED
```

where x is the port number 0 or 1.

___ 2. Repeat the procedure until successfully connected.

If the CC545 is the primary console, successful connection of a secondary console is indicated when only the right screen displays appear on the CC545. If the CC545 was the primary console, but both primary and secondary consoles are reassigned to CDC 721 or 752-compatible terminals, successful connection of these terminals is indicated when the CC545 becomes inactive.

Refer to the CDC 19003 (CC598-A/B) System Console Operations and Maintenance Guide (60463610) for CC598-A/B terminal connection procedures.

CMSE Commands 3

CMSE Commands 3

The common maintenance software executive (CMSE) provides off-line monitor capabilities for the computer system. CMSE monitors the system hardware, loads and initiates execution of maintenance software and utility programs, and loads and initiates execution of microcode as required for the central processing unit (CPU). The CMSE performs the following user functions.

- Transfers data between any two memories.
- Loads programs from a library unit.
- Displays program messages and transfers information in the keyboard buffer to a peripheral processor (PP).
- Displays errors.
- Interprets keyboard commands.
- Places entries in the dayfile.
- Requests and releases I/O channels.
- Requests lines from the display buffer.

Monitoring can be accomplished from a local or remote console. CMSE supports a PP program that interfaces between CMSE and a remote console.

CMSE provides a test capability not feasible in an operating system environment and a common interface for the computer. CMSE also provides display facilities, a keyboard command structure, a loading capability, and diagnostic sequencing. CMSE executes, using the Maintenance Software Library (MSL) 15X, residing on an 834/836, an 844, an 885, or an 895 disk storage device. CMSE also executes, using the MSL 15X, residing on magnetic tape[1] as an alternate or backup device.

CMSE may be initialized following the input/output unit (IOU) deadstart process (short/long deadstart and extended deadstart). This process involves the common test initialization (CTI) sequence, which includes some IOU testing and hardware initialization.

1. Tape operation is not available on CYBER 962/992.

Hardware Requirements

The minimum hardware requirement for CMSE is:

- Three PPs
- Three I/O channels
- One local console (a CC545 connected to channel 10_8, a CC634B or a CC598-A/B console connected to a two-port multiplexer, or a 752-compatible terminal connected to a two-port multiplexer)
- One 639, 66X, 67X, 698, 834/836, 844, 885, or 895 subsystem

CMSE requires three PPs; one each for the I/O driver, the display driver, and the executive program. PP0 is always used. The other two PPs are selectable, using the INITIAL CMSE display shown in figure 2-9.

CMSE also requires three I/O channels to communicate with the I/O driver, the display driver, and the remaining PPs. The I/O driver requires the channel the MSL device is on, and the display driver requires the display channel. CMSE also requires the channel reservation flag for PP communication and program scoping loops.

For the IOU of a CYBER 962 or 992, in addition to the MSL device channel and display channel, channel usage is as follows:

Channel 0	Monitor/IO driver communications
Channel 1	Monitor/display driver communications
Channel 12	Monitor/PP communications
Channel 15 flag	Program scoping loops

The remaining channels and associated channel flags are not used by CMSE on the CYBER 962 or 992 IOU.

Central Memory Usage

CMSE uses channel communications by default. You can select central memory (CM) usage from the initial CMSE display (parameter word 8^2). CMSE reserves a 1-megabyte area in the uppermost part of CM for storage of monitor and I/O driver overlays, pool PP program overlays, communications buffers, a directory of MSL device programs, and page tables and EI. CMSE also protects the reserved area of CM (only the console operator or a pool PP program can use this area on a restricted and controlled basis). Monitor-to-I/O driver communications occur in the protected CM area.

When CMSE is using CM, the channel requirements for CMSE are reduced. Communication between the monitor and the I/O driver through CM means that the channel that was used for these communications is no longer needed.

CM usage can be deselected on the initial CMSE display (parameter word 8^2) if CM is not working. In this case, CMSE would revert to using channel communications between the monitor and I/O driver.

When CM is in use, CMSE performance is increased because monitor and I/O driver overlays are in CM and do not have to be retrieved from the MSL device. Further, the existence of a directory of the programs on the MSL in CM increases CMSE performance, particularly when the MSL device is tape.

When CM is not in use, monitor overlays can still be stored in CM by use of the *OV command.

The monitor controls use of the CMSE area of CM. Keyboard commands which attempt to modify the CMSE area are aborted with an error message unless you unlock the area at the console (refer to UL and LK commands).

The CMSE reserved area of CM is 400,000 octal words (1 megabyte) and is partitioned as shown in figure 3-1.

Word 0 Word 260,000	PP Overlay (260,000 octal words)
Word 270,000	PP Communications Buffer (10,000 octal words)
Word 320,000	CMSE Overlays (30,000 octal words)
Word 340,000	MSL Directory (20,000 octal words)
Word 377,777	Page Tables (40,000 octal words)

Figure 3-1. CMSE Reserved Area of CM

The following paragraphs describe the parts of CMSE area of CM.

NOTE

For dual IOU operation,[2] CMSE uses the last megabyte for IOU 0 and protects the second to last megabyte for use by IOU 1. IOU 1 uses this megabyte and protects the last megabyte for IOU 0. For a single IOU operation, CMSE protects and uses only the last megabyte.

PP Overlay

The PP reserves this part in 10,000-word blocks for use as an overlay or communications area. DEMOT can also reserve it in a 100,000-word block.

PP Communications Buffer

CMSE uses this part to transfer data to and from a pool PP.

CMSE Overlays

CMSE uses this part for CM communications to pool and resident PPs, monitor overlays, I/O driver overlays, and internal CMSE communications.

MSL Directory

CMSE and DEMOT use this part for a directory of the programs on the MSL device.

Page Tables

CMSE uses this part for page tables and EI.

2. Only for systems with an AT478/AT481 IOU as the primary IOU.

CYBER 962, 992 CM Usage

On an AT478/AT481 IOU, CM usage is selected on the initial CMSE display (parameter word 6) when CIO PPs are used. This is not necessary for an AT511/AT512 IOU since internal channel 12 interconnects all PP clusters and is used for PP communications.

Other differences occur between AT478/AT481 and AT511/AT512 IOUs if a dedicated load device is present when CM usage is selected for an AT511/AT512 IOU. In this case, only monitor overlays and not I/O driver overlays are placed in CM. Also, use of command *RV (revert to disk) for an AT511/AT512 IOU allows dropping use of overlays in CM. CMSE automatically executes an *RV command if it cannot locate a desired overlay in CM, and the system displays the informative message CM OVERLAYS CORRUPTED, REVERTING TO DISK on the error lines. Refer to *OV and *RV commands for additional information on AT511/AT512 IOU CM usage.

Command Buffer Editing Utility (CBU)

CMSE uses the CBU to edit command buffers. It employs one of the pool PPs. This PP is unavailable for other use until command buffer editing is terminated by the KE command.

CMSE loads CBU into the PP, and modification of the command buffer takes place within the PP memory. Modification and execution of command buffers is performed with either MSL tape or disk devices.

Software Restrictions

When CMSE uses a magnetic tape for the MSL, the following are not available.

- CMSE dayfile
- The ability to modify existing programs and save these on the library
- The ability to add programs to the library under CMSE

CMSE does not communicate with an idle PP. Communication with an idle PP begins when a CP, TL, LT, EP, HP, MP, or RU command is executed on that PP.

Use of the channel reservation flag for inter-PP communication is not recommended for user programs. If a PP program sets a channel reservation flag, CMSE loses communication with all of the PPs in the system. Communication can be restored only by issuing a UP,num command; where num equals the number of the PP associated with the previously set channel reservation flag. If the channel 7 flag were set by PP5, a UP,7 command must be issued to clear the channel 7 flag.

In order to use peripheral diagnostics from CIO PPs, the CM usage flag 2**0 must be entered. Refer to figure 3-3, the initial CMSE display.

Installation and Initialization

A minimum requirement for an installation tape of MSL 15X consists of a nine-track tape with the binary programs TDL (and overlays), MDL with overlays, DSB, DSP, CEZ, DSQ, MSB, and TDX. Additionally, either MDP with overlays for MSL 151; MEP with overlays for MSL 152; MFP, MF4, MG4, or MG5 with overlays for MSL 153; or MH4 or MH5 with overlays for MSL 155 is required.

The CTI program initializes CMSE.

Keyboard Usage (CC545 only)

When used with CMSE, the CC545 console keyboard keys function as defined within the tabular information of figure 3-2.

Key	Function
Erase (blank, left of =).	Clear keyboard line.
Equal (=).	Switch the B (right screen) display from a test display to a memory display or vice versa.
Backspace (BKSP).	Delete last character entered.
Plus (+).	Increment the base address, index, or ordinal associated with the current A (left screen) display.
Left parenthesis (( ).	Same as (+), for B (right screen) display.
Minus (-).	Decrement the base address, index, or ordinal associated with the current A (left screen) display.
Right parenthesis ( )).	Same as (-), for B (right screen) display.
Forward (blank, right of =).	Execute the command, increment the address parameter, and clear the data parameter. If a data entry is repeated without entering data, no data is stored.
Carriage return (CR).	Execute the command entered. An initial entry of (CR) causes a single byte of zeros for a test command.
Comma, period, or space.	Accepted as a command separator.
Consecutive commas.	When used in a command entry, zeros are entered for the parameter not entered.
Slash (/).	Substitute the last automatically assigned PP number.

CMSE supports the following characters:

A through Z

0 through 9

(	/
)	,
+	.
-	=
*	$

Figure 3-2. CC545 Console Keyboard Usage

Keyboard Usage (CC598-A/B without AEP, CC634B, and 752-Compatible Consoles)

When used with CMSE, the CC598-A/B console without the Advanced Emulation Package (AEP), the CC634B console, or the 752-compatible console keyboard keys function as defined within the tabular information of figures 3-3 and 3-4. These figures do ont contain keyboard usage information for the CC598-A/B consoles with AEP. Refer to figure 3-5.

CC634B	CC598-A/B	752 Compatible	Function
ESC or ERASE	ESC or DEL.	ESC.	Clear keyboard line.
Equal (=) or F3	Equal (=) or F3.	Equal (=).	Switch from a test display to a memory display or vice versa.
Left arrow or CTRL and H.	Left arrow or CTRL and H.	Left arrow or CTRL and H.	Delete last character entered.
Plus (+), left parenthesis (( ), or DOWN.	Plus (+) or left parenthesis .(( )	Plus (+) or left parenthesis .(( )	Increment the base address, index, or ordinal associated with the current display.
Minus (-), right parenthesis ,( )) or UP.	Minus (-) or right parenthesis .( ))	Minus (-) or right parenthesis .( ))	Decrement the base address, index, or ordinal associated with the current display.
Percent (%) or FWD.	Percent (%) or PGDN.	Percent (%).	Execute the command, increment the address parameter, and clear the data parameter. If a data entry is repeated without entering data, no data is stored.
Carriage return (CR) or NEXT.	Carriage return (CR) or NEXT.	Carriage return (CR).	Execute the command entered. An initial entry of (CR) causes a single byte of zeros for a test command.
Comma, period, or space.	Comma, period, or space.	Comma, period, or space.	Accepted as a command separator.
Consecutive commas.	Consecutive commas.	Consecutive commas.	When used in a command entry, zeros are entered for the parameter not entered.
Slash (/).	Slash (/).	Slash (/).	Substitute the last automatically assigned PP number.
Dollar sign ($) or F2.	Dollar sign ($) or F2.	Dollar sign ($).	Select CH/PP display.
Pound sign (#) or F1.	Pound sign (#) or F1.	Pound sign (#).	Select other half of current display.

Figure 3-3. CC598-A/B Without AEP, 721, and 752-Compatible Console Keyboard Usage (Part 1 of 2)

CC634B	CC598-A/B	752 Compatible	Function
CTRL and I.	CTRL and I.	CTRL and I.	Initialize terminal. Manually establish operating characteristics of the console. Used whenever the terminal parameters appear to have been damaged.
F4 or HELP.	F4.	CTRL and Y.	Display an introduction to the information displays and how to access the displays. See HELP display under the description of the Display Help Information (AI, BI) command, later in this section.
CMSE supports the following characters: A through Z a through z (displayed as A through Z) 0 through 9 (/) , + . - = * $ #			
Except as noted above, function keys on the CC634B and CC598-A/B console are not supported. On the CC634B and CC598-A/B consoles, an alarm (beep) sounds when a key other than those shown in this table is pressed.			

Figure 3-4. CC598-A/B Without AEP, 721, and 752-Compatible Console Keyboard Usage (Part 2 of 2)

When used with CMSE, the CC598-A/B console with AEP provides the keyboard funcitons as defined within the tabular information of figure 3-5.

CC598-A/B with AEP	Function
Esc or ERASE EQUAL(=).	Clear keyboard line switch from a test display to a memory display or vice versa.
Left arrow or Ctrl-H.	Delete last character entered.
Plus (+) , PGDN, down arrow.	Increment the base address, index, or ordinal associated with the current A (left screen) display.
Left parenthesis ((), or grey keypad plus (+).	Same as (+) for B (right screen) display.
Minus (-) , PGUP, or up arrow.	Decrement the base address, index, or ordinal associated with the current A (left screen) display.
Right parenthesis ()), or grey keypad minus (-).	Same as (-) for B (right screen) display.
Enter	Command entry.
Esc or Del.	Erase entry
Backspace or left arrow.	Backspace.
Percent (%), insert, or right arrow.	Repeat entry.
F2	Display left screen only.
F3	Display both left and right screens.
F4	Display right screen only.
Comma, period, or space.	Accepted as a command separator.
Consecutive commas.	When used in a command entry, zeros are entered for the parameter not entered.
Slash (/).	Substitute the last automatically assigned PP number.
CMSE supports the following characters: A through Z a through z (displayed as A through Z) 0 through 9	

Figure 3-5. Keyboard Usage For CC598-A/B Console With AEP

Test/Diagnostic Keyboard Commands

To provide operator control of tests and diagnostics, the following commands are used.

Command	Description
S	Stop execution.
R	Restart/reset execution from beginning.
D	Drop program currently executing.
(space)	Start/continue execution.
HELP	For tests and diagnostics, display a brief description of all commands unique to an individual test or diagnostic. Refer to the individual test or diagnostic description for more information; review manuals listed in About This Manual.

A and B Displays (CC598-A/B With AEP and CC545 only)

Both the left and right side (A and B displays) of the CC545 and CC598-A/B screen contain status information. These displays are divided into three parts: the header, the memory or register display area, and the keyboard and message area.

If a secondary console is in use when the CC545 is the primary console, only right screen displays (B displays) appear on the CC545 screen. If both the primary and secondary consoles are CC598-A/B consoles with AEP, then both the left and right screens are supported when both consoles are in use. Refer to the load primary/secondary console driver (LR) command, later in this section.

NOTE

Following a deadstart from a CC598-A/B console, the console is initially supported by DSQ4 (right screen support only). If the CC598-A/B console has the AEP, the user may invoke the LR command to load the dual screen driver (DSD) to utilize both the A and B displays.

A Display

The left side of the screen (A display) contains CPU and PP status. Sample A display headers appear in figures 3-5 through 3-7. The display shows the physical PP numbers if deadstart is from barrel 0 PP0. If deadstart is not from barrel 0 PP0 (for example, barrel 0 PP1 or barrel 2 PP21), the PPs are displayed as logical PP numbers. The PPs used by CMSE are not necessarily 0, 1, and 2, as shown in figures 3-6 and 3-7. Nonexisting or downed PPs do not appear in the display header.

```
       DCPO PP  03  04 05  06  07  10  11          CM IN USE
WXYZ    T*E     D*E P  I   D*  I   F   I           CM   LCKD
IMPP    CP1 PP  20  21 22  23  24  25  26  27  30  31
A=0000  OFF I   I   I  I   I   I   I   I   I   I   I
```

Figure 3-6. A Display Header

```
       DCPO NP  03  04 05  06  07  10  11          CM IN USE
WXYZ    T*E     D*E P  I   D*  I   F   I           CM   LCKD
IMPP    CP1 NP  20  21 22  23  24  25  26  27  30  31
A=0000  OFF I   I   I  I   I   I   I   I   I   I   I
            CP  00  01 02  03  04  05  06  07  10  11
                I   *  F   F   F   F   F   F   F   F
```

Figure 3-7. A Display Header (AT478/AT481 IOU Only)

```
        DCPO PP  03  04 05  06  07  10  11          CM IN USE
WXYZ     T*E     D*E P  I   D*  I   F   I           CM   LCKD
IMPP     CP1 PP  20  21 22  23  24  25  26  27  30  31
         OFF I   I   I  I   I   I   I   I   I   I   I
A=177777
```

Figure 3-8. A Display Header (AT511/AT512 IOU Only)

The following status characters may appear in the A display header.

Character	Description
*	PP or CP is active and in contact with CMSE.
A	PP which was last automatically assigned.
D	PP or CP is requesting test display.
E	PP or CP has identified an error condition.
F	CMSE unable to contact PP in 4096 attempts.
I	PP is idle, available for use, and not in contact with CMSE.
P	PP has parity error status.
IMPP	Displayed when error logging is turned on.
WXYZ or WXY	Indicates clock margins by assigning the following allowable values:

Bit	Description	Allowable Values
W	IOU clock margin	W (wide) or N (narrow)
X	Memory clock margin	W (wide) or N (narrow)
Y	CPU 0 clock margin	W (wide) or N (narrow)
Z	CPU 1 clock margin	W (wide) or N (narrow)

CYBER 96X systems have different clock margin attributes:

Bit	Description	Allowable Values
W	IOU clock margin	W (wide) or N (narrow)
X	CMC and CPU clock margins	W (wide) or N (narrow)
Y	CMC and CPU clock margins	F (fast) or S (slow)

Character	Description
NP	NIO subsystem PP.

Character	Description
CP	CIO subsystem PP. The CPP line and following line are displayed only if the CM IN USE bit is selected on the initial display.
A=	Indicates contents of pseudo A register. A 12-bit value is displayed, except in the case of CYBER 962 and 992 computer systems, where a 16-bit value is displayed.
CP0	CPU 0.
D	Next to CP0 or CP1, the D means that the CPU is the default CPU. All CPU commands that do not specify a CPU number will be applied to the CPU identified with the D. Not applicable to single CPU systems.
T	For dual CPU systems, indicates when a CPU is assigned to the test display. A CPU must be assigned to the test display for its display calls to be accepted. See CU command.
OFF	CMSE is not monitoring for calls in CM. See CN and CF commands.
CM IN USE	Upper part of CM is being used for storage by CMSE during default.
CM LCKD	Upper part of CM in use by CMSE is locked out for modification by keyboard commands. If it reads CM UNLCKD, modification is allowed. Refer to LK and UL commands.

Detailed descriptions of each display can be found in this section.

The PPs that do not exist, and those not available for maintenance, do not appear in the header display.

The keyboard area consists of the bottom five lines of the A display. These lines contain the following information.

Line	Description
Active Command Buffer Name	Contains the name of an active command buffer, if any, issuing commands in place of the keyboard commands.
Keyboard Error	This line may contain an error or informative message (refer to appendix B). For a dual-IOU system[3] only, this line is preceded by an IOU=0 or IOU=1 message.
Keyboard Entry	Contains the type of MSL system and version number followed by the current keyboard entry.
Previous Keyboard Entry	Displays the last keyboard entry accepted.
Error Message	Displays hardware errors (if LE on and errors are encountered).

3. Only pertains to systems with an AT478/AT481 IOU as the primary IOU.

B Display

The right side of the B display screen contains system error and I/O channel status information. Sample B display headers are shown in figures 3-9 and 3-10.

```
CH00  01  02  03  04  05  06  07  10  11  12  13  15  17
 E00 -00  -   F00 -   -10 -   -   F00 -   -   -   P   -
CH20  21  22  23  24  25  26  27  30  31  32  33
 -    -   -   -   F07 -   -   -   -   -   -   -

ST  SS  SB  SC  SE  LE  RT  RS  RB  RC  SM  QL  DR  DE
```

Figure 3-9. B Display Header

```
NCH00  01  02  03  04  05  06  07  10  11  12  13  15  17
  E00 -00  -   F00 -   -10 -   -   F00 -   -   -   P   -
NCH20  21  22  23  24  25  26  27  30  31  32  33
  -    -   -   -   F07 -   -   -   -   -   -   -
CCH00  01  02  03  04  05  06  07  10  11

 ST  SS  SB  SC  SE  LE  RT  RS  RB  RC  SM  QL  DR  DE
```

Figure 3-10. B Display Header (AT478/AT481 IOU Only)

The following information is appended to the B display of CC598-A/B consoles with AEP, when operating with the Dual Screen Driver (DSD).

```
F2 = SELECT LEFT SCREEN
F3 = SELECT LEFT AND RIGHT SCREEN
F4 = SELECT RIGHT SCREEN
```

The following status information may appear in the B display header.

Character	Description
-	Channel inactive when interrogated by CMSE.
F	Channel full when interrogated by CMSE.
E	Channel empty when interrogated by CMSE.
P	Channel contains parity error status. This is displayed only if the error logging for IOU is turned on.
NCH	NIO subsystem channel
CCH	CIO subsystem channel. The CCH line is displayed only if the CM IN USE bit is selected on the initial display.
ST, SS, etc.	Indicates selected parameter conditions. Refer to Program Control Commands later in this section.

A number may follow the hyphen, E, or F channel activity indicators. This number represents the number of the PP for which the channel is reserved. If the number is 00, the channel is reserved for CMSE use.

Console Displays (CC598-A/B without AEP, CC634B, and 752-Compatible)

The A display header and the B display header described for the CC545 displays are combined into a single operator-selectable display on CC634B, 752-compatible, and CC598-A/B[4] consoles. This display, called the CH/PP display is shown in figure 3-11. Pressing **F2** on the CC634B console or entering a dollar sign ($) on a 752-compatible or CC598-A/B console selects this display.

```
CR OFF *CPO  PP  01  04  05  08  07  10  11
        OFF          I   I   I   I   I   I   I
        CP1  PP  20  21  22  23  24  25  26  27  30  31
A=0000  OFF      I   I   I   I   I   I   I   I   I   I

CH00  01  02  03  04  05  06  07  10  11  12  13  15  17
 -   -00 E00 F00 E00 E   E   E   F   E   E   -   -   -
CH20  21  22  23  24  25  26  27  30  31  32  33
 E    E   E   E   E   E   E   E   E   E   E   E




                                                 PAGE 3
```

Figure 3-11. CH/PP Display

The right screen or B displays as described for the CC545 are divided into two 20-line segments for display on CC634B and 752-compatible consoles.

Examples of the first and second halves of the AP or BP command displays as they appear on a CC634B console are shown in figures 3-12 and 3-13.

Typing the pound sign (#) on either keyboard or pressing **F1** on the CC634B keyboard selects the half of the display currently not being shown.

Displays throughout this manual are shown for the CC545, the rules defined above for the CC634B and 752-compatible terminals apply to all CMSE displays.

4. The CC598-A/B console emulates a CDC 721 display terminal.

```
PP00   STATUS 000
-0000  -0010  -0020  -0030  -0040  -0050  -0060  -0070
004252 004016 000220 000000 006376 000000 000000 000000
000036 001020 007777 000030 006376 000000 000000 000000
003600 000000 000036 000000 000000 000024 000000 000000
000001 000000 000052 000000 000000 000000 000000 000000
000000 000014 000024 000000 000003 000000 000000 000004
000000 000015 000000 000003 000001 000010 000000 000002
000000 000000 000000 007776 000000 000000 007777 000024
000000 000060 000000 000000 000000 000000 000060 006376

-0100  -0110  -0120  -0130  -0140  -0150  -0160  -0170
000200 000200 000302 003403 000707 007500 000703 000216
000125 002511 002122 001416 001704 001013 000100 001501
005600 000200 000100 003423 000430 000761 000230 001601
007505 001643 000100 000336 001702 001001 005623 003477
001202 000200 000100 003623 000767 000657 007426 003023
000507 004614 000102 001720 001712 003023 001005 003422
000200 000200 003003 000775 000655 000100 000645 005022
002247 002077 001101 001712 005023 003247 000100 007500

L161   -
BP
                                                  PAGE 1
```

Figure 3-12. AP or BP Command Display (First Half)

```
-0200  -0210  -0220  -0230  -0240  -0250  -0260  -0270
000200 000200 000302 003403 000707 007500 000703 000216
000125 002511 002122 001416 001704 001013 000100 001501
005600 000200 000100 003423 000430 000761 000230 001601
007505 001643 000100 000336 001702 001001 005623 003477
001202 000200 000100 003623 000767 000657 007426 003023
000507 004614 000102 001720 001712 003023 001005 003422
000200 000200 003003 000775 000655 000100 000645 005022
002247 002077 001101 001712 005023 003247 000100 007500

L161   -
BP
                                                  PAGE 2
```

Figure 3-13. AP or BP Command Display (Second Half)

The CC634B console displays also include a definition of the active function keys at the bottom of the screen. Figure 3-14 shows the functions assigned to keys F1 through F4. These functions are defined within the tabular information of figures 3-3 and 3-4.

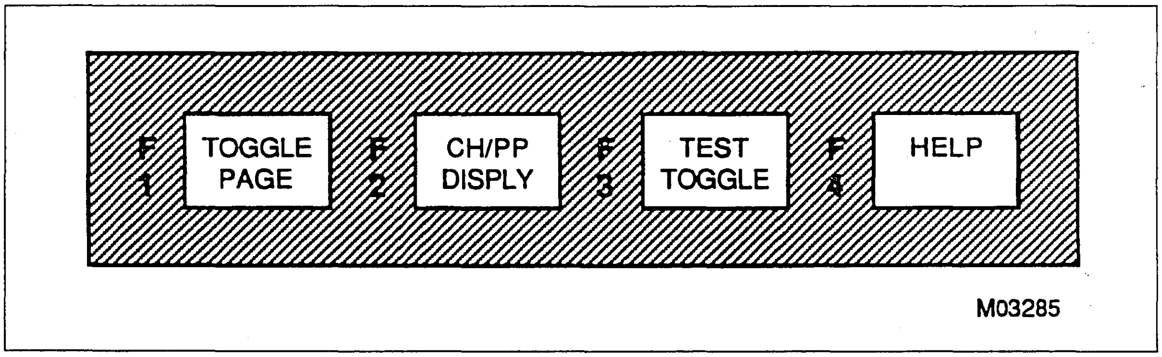

Figure 3-14. Function Key Display (CC634B Console)

Command Types

CMSE contains five types of commands; display, alter, load, execute, and library commands. The system consists of the following functional areas, which use these command types.

- Peripheral processor
- Central memory
- Control store
- Maintenance channel
- Command buffer
- Disk library manipulation

The CMSE commands are two characters in length, except for the library commands and some utility commands, which are two characters prefixed by an asterisk (*).

All keyboard entries are checked for correct format and arguments. If incorrect, the system displays an appropriate error message. Refer to A Display earlier in this section.

Display Commands

The first character of a display command is either an A or B indicating the side of the screen to display the information. For CC598-A/B consoles without the AEP and for CC634B and 752-compatible consoles, the A or B entry is ignored and all entries are treated as B entries. These commands display portions of selected memory, program, register, or library information by writing the information into the A or B display buffer in the display driver.

Alter Commands

The alter commands write information into the selected memory address, register, or microcode location.

Load Commands

The load commands write program information into the selected memory from the MSL peripheral device.

Load from the Common Disk Area

A common disk area (CDA) on the disk device allows the CTI, MSL, and operating systems to read the same versions of system microcode, environment interface (EI), and other programs. The CDA normally exists on all disk devices that contain software for CYBER 180 mainframes. It does not necessarily exist on the sites that choose to install software in manual mode using CTI, CIP, or the tape-to-disk standalone routine (TDX).

CMSE reads only system microcode, peripheral microcode, and EI from the CDA. Other programs and an error log in the CDA are not read or used by CMSE.

System microcodes are programs with 7-character names. The first character is an M, the next four characters are alphanumeric and denote which memory the program is loaded into, and the last two characters are a version level.

Any load request for CMSE to load system microcode (CK, VK, CT, VT, or CS command) where the name of the program to be loaded is specified as four characters, causes CMSE to search a list of all system microcode names for a match between the four characters given on the load request and the second through fifth characters of the microcode names. If there is a match, CMSE searches the CDA for the microcode, and loads the microcode from the CDA if it exists there.

If there is a match but the microcode does not exist in the CDA, CMSE adds an M to the front of the 4-character name given in the load request and searches the MSL library for a program name whose first five characters match the 5-character name including the M. If this second search finds a match, the first program whose name matches is loaded to the memory from the MSL area. If there is no match, CMSE does not search the CDA but attempts to find and load the program with the 4-character name from the MSL area.

Any request to load EI to CM (CC command) causes CMSE to load EI from the CDA if it exists there and from the MSL area if it cannot be found in the CDA.

A load request to CMSE to load peripheral microcode (controlware) via the CW command, where the name of the peripheral microcode file is five characters or less, causes CMSE to search the CDA of the disk for the requested peripheral microcode file. If the file is not found in the CDA, the MSL area is searched. For microcode file names of six characters or greater, only the MSL area of the disk is searched.

Any program which cannot be found in the CDA or the MSL area causes the usual error message to be displayed.

There is no capability within CMSE to write data to the CDA. CTI installs all programs in the CDA.

Execute Commands

The execute commands cause PP or CPU code execution to begin at the selected address.

Library Commands

The library commands read, write, or delete information from the MSL disk device.

Message Displays

Informative messages consist of two parts as shown in figure 3-15. The format of the first part (a) is common to all tests and diagnostics. The format of the second part (b) is optional or unique to each test. The messages appear on line 2 of the test display area (B display). Some tests do not use this format. Refer to manuals containing the individual tests to determine the specific format used.

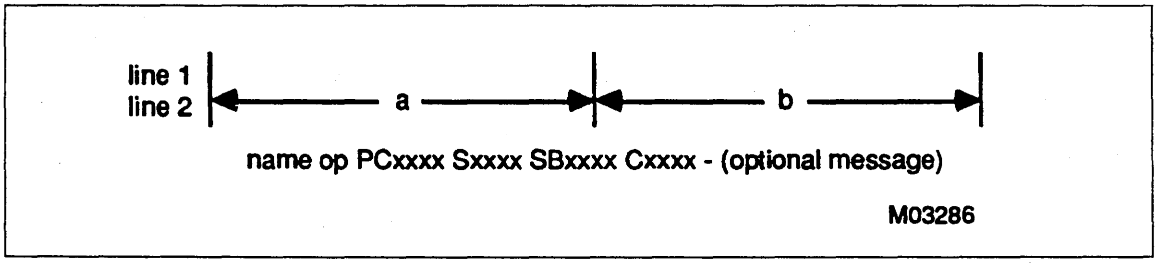

Figure 3-15. Test and Diagnostic Message Display

The message display elements are defined as follows:

name Name of test (four or five-character mnemonic).

op Type of operation performed:

RU	Running message
SMC	Scope mode condition
SMI	Scope mode iteration
SP	Set parameters/stopped for parameters
Sx	Where x is as follows:

C

Stop at condition end

L

Stop at loop

B

Stop at subsection end

S

Stop at section end

T

Stop at test end

E

Stopped on error

I

Stop at iteration end. Report iteration

Rx Where x is as follows:

C

Repeating condition

L

Repeating loop

B

Repeating subsection

S

Repeating section

T

Repeating test

I

Stop at iteration end. Report iteration

PCxxxx Pass count (hexadecimal or decimal).

Sxxxx Current section number (hexadecimal or decimal).

SBxxxx Current subsection number (hexadecimal or decimal).

Cxxxx Current condition number (hexadecimal or decimal).

The optional message area is used for additional information provided by the test.

As each test is loaded, a stop for parameters is done. One of the following displays appear.

```
name SP PCxxxx Sxxxx SBxxxx Cxxxx - PA=x--x yy/mm/dd

              or

name SET PARAMS PA=x--x yy/mm/dd
```

where PA is the parameter buffer address and yy/mm/dd is the year/month/day of the last test revision.

Once a test has started, the system displays a running message as follows:

```
name RU PCxxxx Sxxxx SBxxxx Cxxxx - (optional message)
```

If a test encounters an error condition, the system displays a message along with other information available from the test. The display appears as follows:

```
NAME SE PCxxxx Sxxxx SBxxxx Cxxxx - (optional message)

EC1=pqrs EC2=xxxx TE=xxxx RN=xxxx
```

where the following parameters apply:

EC1	Error code 1.
EC2	Error code 2 (refer to test descriptions).
TE	Total number of errors.
RN	Last random number seed used (optional).

Error code 1 has a set of predefined error codes with the format pqrx defined as follows:

p indicates the failing element as follows:

0	IOU
1	Processor 1
2	Processor 2
3-5	Not used
6	Central memory
7	Shared memory
8	Extended memory
9	Peripherals
A,B	Not used
D	PMF
E	Not used
F	Miscellaneous

q indicates the functional area of the failure as follows:

For p=0

0	Unexpected
1	IOU control
2	PP memory
3	PP register
4	Channel
5	Memory access
6	Operating system bounds register
7	R register
8	Maintenance channel
9-F	Not used

For p=1,
2

0	Cache
1	Map
2	Business data processor (BDP)
3	Arithmetic/Boolean
4	Shift
5	Floating point
6	Multiply
7	Divide
8	Registers
9	Control store
A	Microcode control
B	Instruction fetch
C	Maintenance channel
D	Address control
E,F	Not used

For p=6,
7, or 8

0	Port
1	Distributor
2	Bank
3	Storage unit
4-F	Not used

For p=9

0	Mass storage
1	Magnetic tape
2	Communication/multiplexer
3	Card reader
4	Card punch
5	Line printer
6	Terminal
7	Console
8	Paper tape
9	Optical card reader
A	Plotter
B	SPAM/MAP (sum of products algorithm module/matrix algorithm processor)
C-F	Not used

For p=f

0	Manual operation
1	Information
2	Operator error
3	Undefined error
4-F	Not used

r Indicates the type of failure as follows:

0	Status error
1	Data error
2	Function error
3	Interrupt error
4	Instruction error
5	Address error
6	Error check hardware error
7	Other
8	Multiple failure
9-F	Not used

x Indicates reserved.

For example, EC1=835X indicates an address error on the extended memory storage unit. A code of EC1=F17X indicates an information-only message.

PP Commands

The following commands pertain to PPs. The deadstart PP (DP) command must be executed on a PP which has failed (status F) before that PP can act upon any other command except the SELECT PP DUMP display (AU) command.

CMSE does not communicate with an idle PP. Communication with an idle PP begins when a CP, TL, LT, EP, HP, MP, or RU command is executed on that PP. PP command parameters and displays are in octal format except for the PP and EH commands which use octal format for the address and hexadecimal for the data.

For an AT478/AT481 IOU only, the PP commands must use different formats to select NIO and CIO subsystems. Use xx or Nxx for num to select a PP in an NIO subsystem. Use Cxx for num to select a PP in a CIO subsystem.

For an AT511/AT512 IOU only, the PP commands may use xx, Nxx, or Cxx for num to select a PP. The N or C is significant only for PP load commands CP, LT, and TL.

Peripheral Equipment Tests

The following commands execute peripheral equipment tests on CYBER models 810, 815, 825, 830, 835, 840, 845, 850, 860, 990 and on CYBER 960, 962, 992, and 994 computer systems. To run the peripheral equipment tests, use the following CMSE commands.

Command	Description
LT,name,,p0-p17	Load the test specified by the name parameter into the first available PP at the test load address. Store optional parameters p0 through p17 in the PP memory. Assign the test display to the PP and start the PP at address 100_8.
TL,name,,p0-p17	Load the test specified by the name parameter into the first available PP at the test load address. Store overlays in CM at address xx0000, where xx is the PP number that was loaded. Store optional parameters p0 through p17 in the PP memory. Assign the test display to the PP and start the PP at address 101_8.

Automatic PP Assignment

The lowest numbered available PP is automatically assigned to the CP, TL, or LT command when a PP is not specified. A PP is considered available for assignment if it has an I or Idle status. All PPs present on a system default to Idle when CMSE is loaded. If the deadstart PP parameter in ordinal 8 of the left screen initial display (refer to figure 2-2) is cleared prior to bringing up CMSE, the idle package does not load, making all PPs unavailable for automatic assignment.

When automatic assignment occurs on a CP, TL, or LT command, the letter A appears on the left screen header immediately to the left of the PP assigned and the message

```
CP,mne LOADED INTO PPxx
```

appears on the previous keyboard line of the left screen display. If either the AA or BA display is active when the automatic load occurs, the message

```
mne LOADED
```

appears next to the PP number which was loaded.

After an automatic assignment has occurred, any subsequent PP command to the assigned PP may be made substituting a / for the PP number. For example, the command

```
EP,/,addr,data
```

causes CMSE to load data into the automatically assigned PP.

If a "/" is used and a PP has not been automatically assigned, the message

```
NOT AUTO ASSIGN
```

appears on the error line. The automatically assigned PP number only changes when another automatic assignment is made.

PP Display Commands

The following are the display commands associated with PPs.

Display PP Memory (AP, BP, PP)

The AP and BP command writes 300_8 words of PP memory into the specified display buffer in octal. The PP command accepts the same parameter entries as the AP or BP commands, but displays memory in hexadecimal format. Addressing remains in octal format and the display uses the B display buffer.

Command format:

```
AP,num,fwa or AP,/,fwa

           or

BP,num,fwa or BP,/,fwa

           or

PP,num,fwa or PP,/,fwa
```

where the following parameters apply:

num	One- or two-digit PP number.
/	Substitute the last automatically assigned PP number
fwa	The first word address of memory to be displayed; 0-7777_8 on a 4K PP or 0-17777_8 on an 8K PP. If omitted, the previous fwa value is used. If no previous fwa exists, 0 is assumed.

Examples of AP COMMAND and PP COMMAND displays as they appear (below the A or B headers) on a CC545 or a CC598-A/B console with the AEP are shown in figures 3-16 and 3-17, respectively.

```
PP05 STATUS 000

-0000  -0010  -0020  -0030  -0040  -0050  -0060  -0070
007643 000100 007777 007777 007777 007777 007777 007777
007643 000000 007777 007777 007777 007777 007777 007777
007305 000000 007777 007777 007777 007777 007777 007777
000000 007777 007777 007777 007777 007777 007777 007777
000000 000000 007777 007777 007777 007777 007777 007777
000000 007777 007777 007777 007777 007777 007777 007777
000000 007777 007777 007777 007777 007777 007777 007777
000000 007777 007777 007777 007777 007777 007777 007777

-2200  -2210  -2220  -2230  -2240  -2250  -2260  -2270
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777

-2300  -2310  -2320  -2330  -2340  -2350  -2360  -2370
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777
007777 007777 007777 007777 007777 007777 007777 007777
```

Figure 3-16. AP Command Display

```
PP05 STATUS 000

-0000  -0010  -0020  -0030  -0040  -0050  -0060  -0070
 0FA3   0000   0000   0FFF   0FFF   0FFF   0FFF   0FFF
 0F9E   00C0   0FFF   0FFF   0FFF   0FFF   0FFF   0FFF
 0401   0410   0FFF   0FFF   0FFF   0FFF   0FFF   0FFF
 0000   0000   0FFF   0FFF   0FFF   0FFF   0FFF   0FFF
 0000   0F21   0FFF   0FFF   0FFF   0FFF   0FFF   0FFF
 0E4A   0E41   0FFF   0FFF   0FFF   0FFF   0FFF   0FFF
 0000   0000   0FFF   0FFF   0FFF   0FFF   0FFF   0FFF
 00C0   0F20   0FFF   0FFF   0FFF   0FFF   0FFF   0FFF

-0100  -0110  -0120  -0130  -0140  -0150  -0160  -0170
 0D51   0080   004B   0609   060C   0340   0080   0400
 004A   0067   0708   0E91   0E91   0E51   004B   00A9
 0DD1   0040   00F7   0608   0D91   0000   060A   0709
 004A   0040   0040   0E91   0062   0040   0909   0080
 0E11   00F3   004A   060A   0F71   004A   0789   004B
 0105   0400   0305   0E91   00ED   050D   04C0   060A
 0B00   0040   03C1   060B   0040   030D   0040   0909
 0049   0B00   017E   0E91   004A   0709   0178   0789

-0200  -0210  -0220  -0230  -0240  -0250  -0260  -0270
 04C1   070B   00C3   0809   0175   006E   0FFF   0FFF
 0000   06CA   0040   070B   060A   0FFF   0FFF   0FFF
 0178   070C   004D   06CA   04C0   0FFF   0FFF   0FFF
 030D   0146   0400   070C   0FFF   0FFF   0FFF   0FFF
 0709   0789   00A9   0174   070A   0FFF   0FFF   0FFF
 0080   04C0   0709   0789   0142   0FFF   0FFF   0FFF
 004B   0040   0080   04C1   00EA   0FFF   0FFF   0FFF
 0809   0175   004B   0000   0040   0FFF   0FFF   0FFF
```

Figure 3-17. PP Command Display

Change PP Display Block (DI)

This command allows selection of the first address of each of the three 100-word sections of PP memory written to the display buffer.

Command format:

```
DI,z,fwa
```

where the following parameters apply:

z	The number of the block to be changed, 0 through 5. The A display blocks are numbered 0, 1, and 2. The B display blocks are numbered 3, 4, and 5.
fwa	The first word address of the display blocks; $0\text{-}7777_8$ on a 4K PP or $0\text{-}17777_8$ on an 8K PP. If omitted, zero is assumed.

Select PP Dump Display (AU, BU)

This command dead-dumps 300_8 words of PP memory to the display buffer, using the hardware capability, enabling the PP to output the content of the PP on a selected channel. This command differs from the PP display commands in that these commands require a program to be running in the PP in communication with CMSE. The PP is in an Idle state following execution of this command.

Command format:

```
AU,num,fwa

  or

BU,num,fwa
```

where the following parameters apply:

num	One- or two-digit PP number.
fwa	The first word address of memory to be displayed; $0\text{-}7777_8$ on a 4K PP or $0\text{-}17777_8$ on an 8K PP. If omitted, zero is assumed.

An example of the AU COMMAND display as it appears below the A header is shown in figure 3-18.

```
DUMP PPU 05

 0000   0010   0020   0030   0040   0050   0060   0070

000003 000100 007777 007777 007777 007777 007777 007777
001500 000000 007777 007777 007777 007777 007777 007777
007305 000000 007777 007777 007777 007777 007777 007777
000000 007777 007777 007777 007777 007777 007777 007777
000000 000000 007777 007777 007777 007777 007777 007777
000000 007777 007777 007777 007777 007777 007777 007777
000000 007777 007777 007777 007777 007777 007777 007777
000000 007777 007777 007777 007777 007777 007777 007777

 0100   0110   0120   0130   0140   0150   0160   0170

006525 000200 000113 003011 003014 001500 000200 002000
000112 000147 003410 007225 007225 007125 000113 000251
006725 000100 000367 003010 006625 000000 003012 003411
000112 000100 000100 007225 000142 000100 004411 000200
007025 000363 000112 003012 007565 000112 003611 000113
000405 002000 001405 007225 000355 002415 002300 003012
005400 000100 001701 003013 000100 001415 000100 004411
000111 005400 000576 007225 000112 003411 000570 003611

 0200   0210   0220   0230   0240   0250   0260   0270

002301 003413 000303 004011 000565 000156 007777 007777
000000 003312 000100 003413 003012 007777 007777 007777
000570 003414 000115 003312 002300 007777 007777 007777
001415 000506 002000 003414 007777 007777 007777 007777
003411 003611 000251 000564 003412 007777 007777 007777
000200 002300 003411 003611 000502 007777 007777 007777
000113 000100 000200 002301 000352 007777 007777 007777
004011 000565 000113 000000 000100 007777 007777 007777
```

Figure 3-18. AU Command Display

PP Alter Commands

The following are the alter commands associated with PPs.

Enter PP Memory (EP)

This command enters octal data into a PP at the address specified.

Execution of this command causes CMSE to begin communicating with the designated PP, making it unavailable for automatic assignment.

Command format:

```
EP,num,adrs,data or
EP,/,adrs,data
```

where the following parameters apply:

num One- or two-digit PP number.

/ Substitute the last automatically assigned PP number.

adrs The octal address where the data is to be entered. On a 4K PP memory the range is 5-7577_8, on an 8K PP memory the range is 5-17377_8.

data The right-justified octal data to be entered (one to six octal digits).

Enter PP Memory Hexadecimal Data (EH)

This command enters hexadecimal data into a PP at the address specified.

Execution of this command causes CMSE to begin communicating with the designated PP, making it unavailable for automatic assignment.

Command format:

```
EH,num,adrs,data or
EH,/,adrs,data
```

where the following parameters apply:

num One- or two-digit PP number.

/ Substitute the last automatically assigned PP number.

adrs The octal address where the data is to be entered. On a 4K PP memory the range is 5-7577_8, on an 8K PP memory the range is 5-17377_8.

data The right-justified hexadecimal data to be entered (one to four hexadecimal characters).

Clear PP Memory (KP)

This command enters data into a PP from the first word address to the last word address. If the command is entered with only the num parameter, memory is cleared from location 2 to the beginning of PP resident.

Command format:

```
KP,num,fwa,lwa+1,data,incr or KP,/,fwa,lwa+1,data,incr

                    or

KP,num or KP,/
```

where the following parameters apply:

num	One- or two-digit octal PP number.
/	Substitute the last automatically assigned PP number.
fwa	The first word address of the PP memory; $0\text{-}7577_8$ on a 4K PP, $0\text{-}17377_8$ on an 8K PP.
lwa+1	The last word address plus one of the PP memory; max address is 7600_8 on a 4K PP and 17400_8 on an 8K PP.
data	The right-justified data to be entered (one to six characters). If no data is entered, memory is set to zero.
incr	Amount data is incremented for successive PP words.

Move PP Memory (MP)

This command copies the contents of a source PP to a designated PP from location 2 to the beginning of PP resident.

Execution of this command causes CMSE to begin communicating with the designated PP, making it unavailable for automatic assignment.

Command format:

```
MP,spp,dpp

    or

MP,/,dpp
```

where the following parameters apply:

spp	The source PP number (one or two digits).
/	Substitute the last automatically assigned PP number.
dpp	The destination PP number (one or two digits).

PP Load Command

The following load command is associated with PPs.

NOTE

Because of the requirement to make program length end on 12-, 16-, 60-, 64-bit word boundaries, up to 60 bits of data may appear at the end of a program loaded into memory.

Load PP Program from MSL Device (CP)

This command loads the selected program into a PP from the MSL device. For 12-bit PP programs, parameters p0 through p17 replace program code beginning at PP memory location 5, if used. For 16-bit PP programs, parameters p0 through p17 replace program code beginning at the address in location 5 after the program is loaded. CMSE detects 16-bit programs by examining the type code in the program 7700 table for an H display code.

If you do not specify the PP number with this command, the first available (idle) PP is automatically assigned. An A appears immediately to the left of the assigned PP number in the left screen header.

If you enter no parameters after the program name, the program is loaded, beginning with the address specified in the program table.

Command format:

```
CP,num,name,fwa,p0-p17 or CP,name,fwa,p0-p17

     or

CP,num,name,,p0-p17 or CP,name,,p0-p17

     or

CP,num,name or CP,name

     or

CP,Nxx,name,fwa,p0-p17 or CP,Cxx,name,fwa,p0-p17 (AT478/AT481 IOU only)
```

where the following parameters apply:

num	One- or two-digit PP number.
name	The program name (one to seven alphanumeric characters).
fwa	The first word address of the program.
p0-p17	Optional test parameters stored starting at location 5 of PP memory for 12-bit programs and at the address contained in location 5 of PP memory for 16-bit programs. Commas or spaces must separate parameters. Each parameter is a four-digit octal value for a 12-bit PP load, or a six-digit octal value for a 16-bit PP load.

Nxx Selects PP xx in an NIO subsystem of an AT478/AT481 IOU. If only N is supplied, the first available NIO PP is automatically assigned.

Cxx Selects PP xx in a CIO subsystem of an AT478/AT481 IOU. If only C is supplied, the first available CIO PP is automatically assigned.

NOTE

When the CP, LT, or TL command uses Cxx to select a PP in an AT511/AT512 IOU, the PP load is performed as if the PP is a CIO PP in an AT478/AT481 IOU.

Call Test into PP and Use Library Overlays (LT)

This command loads the test specified by the name parameter to a PP at address fwa. The test display is assigned to the specified PP and the PP is started at address 100_8.

If you do not specify the PP number with this command, the first available (idle) PP is automatically assigned. An A appears immediately to the left of the assigned PP number in the PP display.

Command format:

```
LTxx,name,fwa,p0-p17 or LT,name,fwa,p0-p17

     or

LT,Nxx,name,fwa,p0-p17 or LT,Cxx,name,fwa,p0-p17 (AT478/AT481 IOU only)
```

where the following parameters apply:

xx One- or two-digit PP number.

name Name of test (three or four characters).

fwa First word address (default is test load address).

p0-p17 Optional test parameters to be stored starting at location 5 of PPxx memory for 12-bit programs and at the address contained in location 5 of PP memory for 16-bit programs. Each parameter is a four-digit octal value for a 12-bit PP load, or a six-digit octal value for a 16-bit PP load. Commas or spaces must separate parameters. No punctuation indicates no parameters are stored. These parameters overide parameters in the library file.

Nxx Selects PP xx in an NIO subsystem of an AT478/AT481 IOU. If only N is supplied, the first available NIO PP is automatically assigned.

Cxx Selects PP xx in a CIO subsystem of an AT478/AT481 IOU. If only C is supplied, the first available CIO PP is automatically assigned.

Call Test into PP and Use CM Overlays (TL)

This command loads the test specified by the name parameter to a PP at address fwa. The test display is assigned to the specified PP and the PP is started at address 101_8.

If you do not specify the PP number with this command, the first available (idle) PP is automatically assigned. An A appears immediately to the left of the assigned PP number in the PP display.

This entry is for loading tests when running from a tape library.

Command format:

```
TLxx,name,fwa,p0-p17 or TL,name,fwa,p0-p17

     or

TL,Nxx,name,fwa,p0-p17 or TL,Cxx,name,fwa,p0-p17 (AT478/AT481 IOU only)
```

where the following parameters apply:

xx	One- or two-digit PP number. Also, the first two digits of the CM address into which overlays are loaded.
name	Name of test (three or four alphanumeric characters).
fwa	First word address; defaults to test load address.
Nxx	Selects PP xx in an NIO subsystem of an AT478/AT481 IOU. If only N is supplied, the first available NIO PP is automatically assigned.
Cxx	Selects PP xx in a CIO subsystem of an AT478/AT481 IOU. If only C is supplied, the first available CIO PP is automatically assigned.
p0-p17	Optional test parameters to be stored starting at location 5 of PPxx memory for 12-bit programs and at the address contained in location 5 of PP memory for 16-bit programs. Each parameter is a four-digit octal value for a 12-bit PP load or a six-digit octal value for a 16-bit PP load. When fewer than four digits are specified, data enters right-justified. Commas or spaces must separate the parameters. No punctuation indicates that no parameters are stored. These parameters override parameters in the library file.

PP Execute Commands

The following execute commands are associated with PPs.

Deadstart PP (DP)

This command performs a software deadstart of a PP using the hardware capability to place a PP in a deadstart condition on a specified channel. It then loads the PP resident program into the PP address locations 7600_8 through 7777_8 on a 4K PP, or 17400_8 through 17777_8 on an 8K PP and starts executing in the idle loop portion of that program. This command releases any channels reserved by this PP and makes the PP available for automatic assignment (status is Idle).

NOTE

If the PP selected has reserved channel 17, the maintenance channel (MCH), it is possible for this command to hang the IOU, or cause other unpredictable results and may require a hardware deadstart.

Command format:

```
DP,num or DP,/
```

where the following parameters apply:

num	One- or two-digit PP number.
/	Substitute the last automatically assigned PP number.

Execute PP Program (RU)

This command causes the selected PP to begin executing instructions at the address specified.

Execution of this command causes CMSE to begin communicating with the designated PP, making it unavailable for automatic assignment.

Command format:

```
RU,num,adrs or RU,/,adrs
```

where the following parameters apply:

num	One- or two-digit PP number.
/	Substitute the last automatically assigned PP number.
adrs	The address where program execution starts; 0-7577_8 on a 4K PP or 0-17377_8 on an 8K PP.

The following two commands are special purpose RU commands that control individual tests.

```
RUx,100 Start or continue PPx, and load overlays from the library.

RUx,101 Start PPx putting overlays in central memory.
```

Halt PP Program Execution (HT)

This command causes the selected PP to remain in the idle loop portion of the resident code the next time resident code is executed. Programs halted using this command can usually be continued by entering an RU command with an adrs parameter value of 7603_8 on a 4K PP or 17603_8 on an 8K PP.

Command format:

```
HT,num or HT,/
```

where the following parameters apply:

num	One- or two-digit PP number.
/	Substitute the last automatically assigned PP number.

Repetitive Deadstart (RP)

This command uses the PP deadstart capability to repetitively download up to 10 PP instructions to a selected PP, and allows enough time for each instruction to execute before repeating the process. This command remains in the keyboard buffer and executes until another keyboard entry is made.

Command format:

```
RP,num,p1,p2,...,p10
```

where the following parameters apply:

num	One- or two-digit PP number.
p1-p10	One to 10 PP instructions to be executed during each deadstart. Each parameter is one instruction word in machine language and is a four- digit octal value for a 12-bit PP load, or a six-digit octal value for a 16-bit PP load.

PP Library Command

The following library command is associated with PPs.

Write PP Program to MSL Disk (*WP)

This command allows the contents of the PP, beginning with the first word address and continuing to the last word address, to be written on the MSL disk. The PP programs written to disk using this command are always written as 16-bit programs.

Command format:

```
*WP,base,num,name,fwa,lwa+1
```

where the following parameters apply:

base — Indicates the data format as follows:

O	Octal
H	Hexadecimal
Omitted	Hexadecimal

num — One- or two-digit PP number.

name — The name of the program to be written to disk (one to seven characters).

fwa — The first word address of the program to be written to disk; $0\text{-}7577_8$ on a 4K PP or $0\text{-}17377_8$ on an 8K PP. If omitted, zero is assumed.

lwa+1 — The last word address plus one of the program to be written to disk. If omitted, 7600_8 for a 4K PP and 17400_8 for an 8K PP is used.

CM Commands

The following commands pertain to CM. Except for the display commands, the CM commands default to hexadecimal format for all address and data entries. For octal entries, the base parameter must be used. All numerical entries following the base parameter are in that base. The two modes (octal and hexadecimal) may not be mixed.

CM Display Commands

The following display commands are associated with CM.

Display Octal CM (AC, BC, AD, BD)

This command causes 40_8 words of CM to be written in the display buffer.

Command format:

```
AC,fwa or AC,fwa1,fwa2

  or

BC,fwa

  or

AD,fwa

  or

BD,fwa
```

where the following parameters apply:

fwa	The first word (absolute) address of memory to be displayed in octal.
fwa1,fwa2	The first word (absolute) address of memory to be displayed in octal. When two first words are specified, 20_8 words are displayed for each word.
C	Indicates four columns of five octal digits each are to be displayed.
D	Indicates five columns of four octal digits each are to be displayed.

Examples of AC COMMAND and AD COMMAND displays are respectively shown in figures 3-19 and 3-20.

```
AC*00000000

00000000   00010   00000   00000   00130
00000001   00050   00400   04000   40134
00000002   00110   01000   10001   00140
00000003   00150   01400   14001   40144
00000004   00210   02000   20002   00150
00000005   00250   02400   24002   40154
00000006   00310   03000   30003   00160
00000007   00350   03400   34003   40164

00000010   00410   04000   40004   00170
00000011   00450   04400   44004   40174
00000012   00510   05000   50005   00200
00000013   00550   05400   54005   40204
00000014   00610   06000   60006   00210
00000015   00650   06400   64006   40214
00000016   00710   07000   70007   00220
00000017   00750   07400   74007   40224
```

Figure 3-19. AC Command Display

```
AD*00001000

   00   0020   0000   0000   0000   0020
   01   0000   0000   0000   0000   0000
   02   0020   0000   0002   0000   0030
   03   0000   0000   0000   0000   0000
   04   0020   0000   0004   0000   0040
   05   0000   0000   0000   0000   0000
   06   0020   0000   0006   0000   0050
   07   0000   0000   0000   0000   0000

   10   0020   0000   0010   0000   0060
   11   0000   0000   0000   0000   0000
   12   0020   0000   0012   0000   0070
   13   0000   0000   0000   0000   0000
   14   0020   0000   0014   0000   0100
   15   0000   0000   0000   0000   0000
   16   0020   0000   0016   0000   0110
   17   0000   0000   0000   0000   0000
```

Figure 3-20. AD Command Display

Display Hexadecimal CM, Byte Address (AB, BB)

This command causes 20 hexadecimal words of CM to be written to the display buffer with real memory byte addresses, beginning with the CM word containing the first byte address.

Command format:

```
AB,fba or AB,fba1,fba2

  or

BB,fba
```

where the following parameters apply:

fba	First byte address of CM written to the display buffer. If omitted, the previous fba value is used. If no previous fba value exists, zero is assumed.
fba1,fba2	First byte address of CM written to the display buffer. When first two words are specified, 10 hexadecimal words are displayed for each word.

Examples of the AB COMMAND and BB COMMAND displays as they appear below the A and B headers are shown in figures 3-21 and 3-22, respectively.

```
AB*0007000

0007000 1AFF 3D00 E948 38C0
0007008 00AB 06C0 00AB 07C0
0007010 A900 0020 1801 DF31
0007018 0098 3D01 1AFF 3D00
0007020 E948 38C0 00AB 06C0
0007028 0019 07C0 A900 0020
0007030 1801 DF31 00A0 3D01
0007038 1AFF 3D00 E948 38C0

0007040 0019 07C0 00AB 06C0
0007048 A900 0020 1801 DF31
0007050 00A8 3D01 2E0F 0000
0007058 0000 0000 0000 0000
0007060 0000 0000 0000 0000
0007068 0000 0000 0000 0000
0007070 0000 0000 0000 0000
0007078 0000 0000 0000 0000

0007080 0000 0000 0000 0000
0007088 0000 0000 0000 0000
0007090 0000 0000 0000 0000
0007098 0000 001F 4000 0000
00070A0 0000 001F 4000 0000
00070A8 0000 001F C000 0000
00070B0 0000 0000 0000 0000
00070B8 0000 0000 0000 0000

00070C0 0000 0000 0000 0000
00070C8 0000 0000 0000 0000
00070D0 0000 0000 0000 0000
00070D8 0000 0000 0000 0000
00070E0 0000 0000 0000 0000
00070E8 0000 0000 0000 0000
00070F0 0000 0000 0000 0000
00070F8 0000 0000 0000 0000
```

Figure 3-21. AB Command Display

```
BB*0007000

0007000 1AFF 3D00 E948 38C0
0007008 00AB 06C0 00AB 07C0
0007010 A900 0020 1801 DF31
0007018 0098 3D01 1AFF 3D00
0007020 E948 38C0 00AB 06C0
0007028 0019 07C0 A900 0020
0007030 1801 DF31 00A0 3D01
0007038 1AFF 3D00 E948 38C0

0007040 0019 07C0 00AB 06C0
0007048 A900 0020 1801 DF31
0007050 00A8 3D01 2E0F 0000
0007058 0000 0000 0000 0000
0007060 0000 0000 0000 0000
0007068 0000 0000 0000 0000
0007070 0000 0000 0000 0000
0007078 0000 0000 0000 0000

0007080 0000 0000 0000 0000
0007088 0000 0000 0000 0000
0007090 0000 0000 0000 0000
0007098 0000 001F 4000 0000
00070A0 0000 001F 4000 0000
00070A8 0000 001F C000 0000
00070B0 0000 0000 0000 0000
00070B8 0000 0000 0000 0000

00070C0 0000 0000 0000 0000
00070C8 0000 0000 0000 0000
00070D0 0000 0000 0000 0000
00070D8 0000 0000 0000 0000
00070E0 0000 0000 0000 0000
00070E8 0000 0000 0000 0000
00070F0 0000 0000 0000 0000
00070F8 0000 0000 0000 0000
```

Figure 3-22. BB Command Display

Display Hexadecimal CM (AH, BH)

This command causes 20 hexadecimal words of CM to be written to the display buffer.

Command format:

```
AH,fwa or AH,fwa1,fwa2

  or

BH,fwa
```

where the following parameters apply:

fwa	The first word address (absolute) of CM to be displayed in hexadecimal. If omitted, the display starts either at the address previously entered or at zero.
fwa1,fwa2	The first word address (absolute) of CM to be displayed in hexadecimal. When first two words are specified, 10 hexadecimal words are displayed for each word.

An example of the AH COMMAND display is shown in figure 3-23.

```
AB*0007000

000200   9010   0000   0000   0010
000201   0000   0000   0000   0000
000202   9010   0000   0200   0018
000203   0000   0000   0000   0000
000204   9010   0000   0400   0020
000205   0000   0000   0000   0000
000206   9010   0000   0600   0028
000207   0000   0000   0000   0000

000208   9010   0000   0800   0030
000209   0000   0000   0000   0000
00020A   9010   0000   0A00   0038
00020B   0000   0000   0000   0000
00020C   9010   0000   0000   0040
00020D   0000   0000   0000   0000
00020E   9010   0000   0E00   0048
00020F   0000   0000   0000   0000

000210   9010   0000   1000   0050
000211   0000   0000   0000   0000
000212   0000   0000   0000   0000
000213   0000   0000   0000   0000
000214   0000   0000   0000   0000
000215   0000   0000   0000   0000
000216   0000   0000   0000   0000
000217   0000   0000   0000   0000

000218   0000   0000   0000   0000
000219   0000   0000   0000   0000
00021A   0000   0000   0000   0000
00021B   0000   0000   0000   0000
00021C   0000   0000   0000   0000
00021D   0000   0000   0000   0000
00021E   0000   0000   0000   0000
00021F   0000   0000   0000   0000
```

Figure 3-23. AH Command Display

Display Virtual Memory (AV, BV)

This command causes up to 20 hexadecimal words of CM to be displayed. Parameter values not supplied in the command will be read from the process state registers unless otherwise noted. The RMA value displayed is adjusted to a byte address ending in zero or eight. An asterisk (*) is displayed above the actual starting byte. The increment(+)/decrement(–) keys adjust the displayed PVA and cause a new PVA to RMA translation. Real memory data is displayed only for valid PVAs. If PVA to RMA translation is not possible, an error message indicating the invalid condition appears below the display header.

NOTE

Only one V display at a time is allowed (AV or BV). If a second V command is entered, the current V display becomes a BYTE ADDRESS CM display.

Command format:

```
AV,pva,sta,pta,psm,ptl,stl
AV,Pnum,pva,J
AV,Pnum,pva,M
AV,Pnum,pva
AV,Pnum

        or

BV,pva,sta,pta,psm,ptl,stl
BV,Pnum,pva,J
BV,Pnum,pva,M
BV,Pnum,pva
BV,Pnum
```

where the following parameters apply:

num	Optional CPU select code for obtaining Process State Register values. If omitted default CPU is assumed.
pva	Optional processor virtual address (12 hexadecimal digits).
sta	Optional segment table address (eight hexadecimal digits).
pta	Optional page table address (eight hexadecimal digits).
psm	Optional page size mask (7 bits).
ptl	Optional page table length (four hexadecimal digits). The ptl is 8 bits, except for model 960 where it is 14 bits.
stl	Optional segment table length (three hexadecimal digits).
J	sta and stl are obtained from the Job Process Exchange package. Remaining parameter values obtained from Process State Registers.
M	sta and stl are obtained from the Monitor Process Exchange package. Remaining parameter values obtained from Process State Registers.

An example of the BV COMMAND display as it appears below the B header is shown in figure 3-24.

```
PVA=B00740000000 STA=B00002400 PTA=00001000 PSM=00
                 STL=009       PTL=0000
                 *
RMA   00021C00     0000 0000 0002 1C00
      00021C08     FFFF FFFF FFFF FFFF
          :                       :
      00021CF8     FFFF FFFF FFFF FFFF
```

Figure 3-24. BV Command Display

Display CM in Exchange Package Format (AX, BX)

This command displays the contents of an area of CM beginning at fwa in exchange package format.

Command format:

```
AX,fwa

  or

BX,fwa
```

where the following parameter applies:

fwa	First word (absolute) address of CM to be displayed in octal.

An example of the AX COMMAND display as it appears below the A header is shown in figure 3-25.

```
AX*001000

00  -P-  00  001300  000000  000000
01  RA-  00  000000  000000  000000
02  FL-  00  777777  000000  000000
03  EM-  00  200000  000000  000000
04  RAE  00  000000  000000  000000
05  FLE  77  777777  000000  000000
06  MA-  00  001000  000000  000000
07       00  000000  000000  000000

10  0000     0000    0000    0000    0000
11  0000     0000    0000    0000    0000
12  0000     0000    0000    0000    0000
13  0000     0000    0000    0000    0000
14  0000     0000    0000    0000    0000
15  0000     0000    0000    0000    0000
16  0000     0000    0000    0000    0000
17  0000     0000    0000    0000    0000

20  -P-  00  001336  000000  000000
21  RA-  00  000000  000000  000000
22  FL-  00  777777  000000  000000
23  EM-  00  200000  000000  000000
24  RAE  00  000000  000000  000000
25  FLE  77  777777  000000  000000
26  MA-  00  001020  000000  000000
27       00  000000  000000  000000

30  0000     0000    0000    0000    0000
31  0000     0000    0000    0000    0000
32  0000     0000    0000    0000    0000
33  0000     0000    0000    0000    0000
34  0000     0000    0000    0000    0000
35  0000     0000    0000    0000    0000
36  0000     0000    0000    0000    0000
37  0000     0000    0000    0000    0000
```

Figure 3-25. AX Command Display

CM Alter Commands

The following alter commands are associated with CM.

Enter Byte of CM Data (EB)

This command alters up to one CM word of data, one character at a time, starting at the byte address of the data.

Command format:

```
EB,badrs,data
```

where the following parameters apply:

badrs	The byte address of the data to be altered.
data	The data to be entered, 1 to 16 hexadecimal digits (left-justified). If omitted, there is no change to byte address.

Enter Word of CM Data (EC)

This command enters data into a specified CM address. The mode selected (octal or hexadecimal) is determined by the base parameter. This command does not purge cache if the base is octal.

Command format:

```
EC,base,adrs,data

      or

EC,adrs,data
```

where the following parameters apply:

base — Indicates the data format as follows:

O	Octal
H	Hexadecimal
Omitted	Hexadecimal

adrs — The address where the data is to be entered.

data — The right-justified data (1 to 20 octal or 1 to 16 hexadecimal characters). If omitted, there is no change to CM address.

Set/Clear Block of CM (KC)

This command enters data into CM beginning at the value of the fwa parameter. If the data parameter is omitted, zeros are entered. This command purges cache if the base is octal.

Command format:

```
KC,base,fwa,lwa+1,data,incr
```

where the following parameters apply:

base	Indicates the data format as follows: O Octal H Hexadecimal Omitted Hexadecimal
fwa	The first word address of the data entry.
lwa+1	The last word address plus one of the data entry.
data	The right-justified data (1 to 20 octal or 1 to 16 hexadecimal characters). If omitted, zero is assumed. Any blank space appearing in the data is interpreted as a delimiter and the subsequent data is assumed to be the incr value.
incr	Data increment for successive CM words (1 to 20 octal or 1 to 16 hexadecimal characters).

CAUTION

The KC command fails when bringing up CMSE from tape after an initial power-on. This problem occurs on CYBER 170 Models 815/825, CYBER 180 810/810A, 830/830A computer systems. After a power-on, the following CMSE commands must be executed before CM reads and writes can be performed.

CX,M	Master Clear CMC
CE,I	Clear IOU Errors
CE,M	Clear Memory errors
ER,M,20,0	Clear Memory Environmental Control Register
ER,M,21,0	Clear Memory Bounds Register

If deadstart is from disk, CTI performs the equivalent of the above commands before CMSE is brought into execution. During an initial installation in which you deadstart from tape the above commands should be executed.

Move CM (MC)

This command copies the number of CM words specified by the nmbr parameter beginning at a source address and ending at a destination address. This command purges cache if the base is octal.

Command format:

```
MC,base,sadr,dadr,nmbr
```

where the following parameters apply:

base	Indicates the data format as follows: O Octal H Hexadecimal Omitted Hexadecimal
sadr	The source address where the data is to be copied.
dadr	The destination address or last address where the data is to be copied.
nmbr	The number of words to be copied.

Set A Register Position at CM Address f to d (XA)

This command stores d into bits 35 through 18 (CYBER 170 exchange package A field) of the specified CM word. The other fields of the word are unchanged. When fewer than six octal digits are entered, the data stores in the lower-order bits of the field.

Command format:

```
XAf,d
```

where the following parameters apply:

f	CM address; one to six octal digits. Leading zeros are not required.
d	Value to be set into A register field; as many as six octal digits.

Set B Register Position at CM Address f to d (XB)

This command stores d into bits 17 through 0 (CYBER 170 exchange package B field) of the specified CM word. The other fields of the word are unchanged. When fewer than six octal digits are entered, the data stores in the lower-order bits of the field.

Command format:

```
XBf,d
```

where the following parameters apply:

f	CM address; one to six octal digits. Leading zeros are not required.
d	Value to be set into B register field; as many as six octal digits.

Set P Register Position at CM Address f to d (XP)

This command stores d into bits 59 through 36 (CYBER 170 exchange package P field) of the specified CM word. The other fields of the word are unchanged. When fewer than eight octal digits are entered, the data stores in the lower-order bits of the field. When entering other CP registers (RA, FL, EM, RAE, FLE, and MA) use the XP command and the actual CM address.

Command format:

```
XPf,d
```

where the following parameters apply:

f	CM address; one to six octal digits. Leading zeros are not required.
d	Value to be set into P register field; as many as eight octal digits.

CM Load Command

The following load commands are associated with CM. The programs are loaded 64 bits per word to a hexadecimal address if the type parameter in the program 7700 table is set to H. Otherwise, the program is loaded 60 bits per word to an octal address.

NOTE

Because of padding to make program lengths end on 12-, 16-, 60-, or 64-bit word boundaries, up to 60 bits of data may appear at the end of a program loaded into memory.

Load CM Program from MSL Device (CC)

This command loads the specified program beginning at the specified address.

Command format:

```
CC,name,fwa
```

where the following parameters apply:

name	The program name (one to seven characters).
fwa	The first word address where the program is to be loaded. If omitted, the program is loaded at the address designated in the program tables.

NOTE

Several diagnostic programs are loaded into CM in a compressed format with large contiguous blocks of zeros removed. The CMSE contains a PP utility program that expands these compressed diagnostics to their executable format. The utility program displays an error message if it is unable to expand the diagnostic.

The compressed format and utility program reduce MSL space requirements and the time required to load the diagnostic into CM.

CM Execute Commands

The following execute commands are associated with CM.

Monitor CM for CMSE Calls (CN)

This command causes CMSE to monitor the CM address for communication with a CM based program. A program making a CMSE call starts the call block at this address.

Command format:

```
CN,Pnum,base,adrs

    or

CN,Pnum,base
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
base	Indicates the data format as follows: O Octal H Hexadecimal Omitted Hexadecimal
adrs	The address in CM to be monitored (17 bits maximum). If the adrs parameter is missing, zero is used.

Stop Monitoring CM for CMSE Calls (CF)

This command negates the effect of the CN command. CMSE does not monitor CM for communication with CM based programs.

Command format:

```
CF,Pnum
```

where the following parameter applies:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.

CM Library Commands

The following library commands are associated with CM.

Compare Hexadecimal Data in CM (CH)

This command compares data from two data blocks in CM. The compare is done on one CM word at a time until an error is encountered or the end of memory is reached. At termination of compare, the system displays the number of words compared with no errors on the previous entry line of the A display in the following format.

```
CH,fwa1,fwa2,xxx
```

where the following parameters apply:

xxx	Compare word count or blanks (if end of memory is reached without encountering any compare errors).

Command format:

```
CH,adrs1,adrs2
```

where the following parameters apply:

adrs1	First word address of first data block in CM.
adrs2	First word address of the second data block in CM.

Write CM Program to MSL Disk (*WC)

This command causes the contents of CM to be written to the MSL disk, beginning at fwa and continuing until lwa.

Command format:

```
*WC,base,name,fwa,lwa+1
```

where the following parameters apply:

base	Indicates the data format as follows:	
	O	**Octal**
	H	**Hexadecimal**
	Omitted	**Hexadecimal**
name	The program name (one to seven alphanumeric characters).	
fwa	The first word address of the CM to be written.	
lwa+1	The last word address plus one of the CM to be written.	

CPU Commands

The following commands are related to the CPU.

Assign Job Display to a CPU (CU)

This command informs CMSE which CPU's Call Block Area in CM is to be used to accept display calls for displaying on the test display (B screen). This command also causes the CM C display on the B screen to be displayed then replaced by the display call when one is received. The CN command defines the Call Block location for a CPU-based program.

Command format:

```
CU,Pnum
```

where the following parameter applies:

num	CPU number selection. num is the CPU number.

Assign the Default CPU (PD)

This command assigns the default processor; the CPU number to be used as the default CPU for all subsequent CPU commands that do not specify a CPU number.

Command format:

```
PD,Pnum
```

where the following parameter applies:

num	CPU number selection. num is the CPU number.

Control Store Commands

The following commands are associated with control store (CS). All CS command parameters are in hexadecimal format.

CS Display Command

The following display is associated with CS.

Select CS Display (AK, BK)

This command causes 20 hexadecimal locations of CS to be written to the display buffer.

Command format for CYBER models 810, 815, 825, 830, 835, 840, 845, 850, 855, and 860 only:

```
AK,Pnum,My,fwa

      or

BK,Pnum,My,fwa
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
y	CS memory select for CYBER models 840, 845, 850, 855, 860, and for CYBER 960 only. 0 — Display primary. 1 — Display shadow (except 845 and 855). Omitted — Display the memory selected by shadow select bit (bit 32 of DEC).
fwa	The first word address of the locations of CS to be written (zero to four digits). Default is to whatever value currently exists in the address buffer. For CYBER 960 only, it determines lower or upper areas of CS. Address ranges are: Lower primary (M=0) — 000_{16} to $7FF_{16}$ Upper primary (M=0) — 800_{16} to FFF_{16} Lower shadow (M=1) — 000_{16} to $7FF_{16}$ Upper shadow (M=1) — 800_{16} to FFF_{16}

An example of the AK COMMAND display as it appears under the A header is shown in figure 3-26.

```
AK CPO 000834

000834  3D0000DA E11B0825 350000DB 011AE825
000835  A225311F FAF48900 200001E0 C0070610
000836  C2813810 80109E05 01000000 00000000
000837  C2813810 80109E05 00200000 00000000
000838  C2813810 80109E05 00100000 00000000
000839  C2813810 80109E05 00080000 00000000
00083A  C2813810 80109E05 00040000 00000000
00083B  020000C0 00148000 1382C000 00148000
00083C  820000D2 1EBEAD00 80180000 30000000
00083D  A282C01E B67EA900 80380000 18000000
00083E  828408C0 00148000 018001E0 0004002E
00083F  C7000840 121E1585 04000000 00000000
000840  070000C2 921E6995 0785F01E 121EA981
000841  070000C2 B21EA995 0785F01E 121E6981
000842  A2890012 12034D80 80380000 38000000
000843  C787A800 12035585 03FF0000 00000000
000844  C787C800 12035585 07FF0000 00000000
000845  C787C800 12035585 F8000000 00000000
000846  C768D800 1216D585 00000000 00DEADC6
000847  C768D800 1216D585 00000000 00DEADC7
000848  070000C2 B2034981 0787A012 12036D81
000849  070000C2 B2034981 0787C012 12036D81
00084A  C768D800 1216D585 00000000 0CDEADAA
00084B  A2857002 811E6816 42380020 00D00000
00084C  C7890800 12035585 07FF0000 00000000
00084D  C7890800 12035585 F8000000 00000000
00084E  C7890800 12035585 03FF0000 00000000
00084F  C7890800 12035585 00000000 00000000
000850  A2851000 00148000 000001E0 0004002E
000851  B7853002 811E8816 42380020 01D00000
000852  820000C2 B21EA994 80380000 38000000
000853  B3855002 A11EC816 42380020 01D00000
```

Figure 3-26. AK Command Display for CYBER Models 810, 815,825, 830, 835, 840, 845, 850, 855, 860 and CYBER 960

Command format for model 990 only:

```
AK,Pnum,L,fwa

     or

BK,Pnum,L,fwa
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
L	Logical display. In logical format, bits read from memory are arranged in logical groups for display purposes instead of sequentially as in physical format. Physical format results when L is omitted.
fwa	The first word address of the locations of CS to be written (zero to four digits). Default is to whatever value currently exists in the address buffer.

Examples of physical and logical formats of the AK COMMAND displays as they appear under the A or B header are shown in figures 3-27 and 3-28.

```
AK CP0 000

000     0101 0101 0101 0101 0080 0000 0000 0000
        1818 1818 1818 1818 0000 0000 0000 0000
001     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
002     0101 0101 0101 0101 0440 0000 0000 0004
        1818 1818 1818 1818 0000 0000 0000 0000
003     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
004     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
005     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
006     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
007     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
008     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
009     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
00A     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
00B     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
00C     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
00D     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
00E     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
00F     0000 0000 0000 0000 BFC0 0000 0000 0000
        0000 0000 0000 0000 0000 0000 0000 0000
```

Figure 3-27. AK Command Display for CYBER Model 990

```
AK CPO 000

000     0080 0000 0000 0000 0000 0001 0000 0000
001     BFC0 0000 0000 0000 0000 0000 0000 0000
002     0440 0000 0000 0004 0000 0001 C000 0000
003     BFC0 0000 0000 0000 0000 0000 0000 0000
004     BFC0 0000 0000 0000 0000 0000 0000 0000
005     BFC0 0000 0000 0000 0000 0000 0000 0000
006     BFC0 0000 0000 0000 0000 0000 0000 0000
007     BFC0 0000 0000 0000 0000 0000 0000 0000
008     BFC0 0000 0000 0000 0000 0000 0000 0000
009     BFC0 0000 0000 0000 0000 0000 0000 0000
00A     BFC0 0000 0000 0000 0000 0000 0000 0000
00B     BFC0 0000 0000 0000 0000 0000 0000 0000
00C     BFC0 0000 0000 0000 0000 0000 0000 0000
00D     BFC0 0000 0000 0000 0000 0000 0000 0000
00E     BFC0 0000 0000 0000 0000 0000 0000 0000
00F     BFC0 0000 0000 0000 0000 0000 0000 0000
010     BFC0 0000 0000 0000 0001 0000 0000 0000
011     BFC0 0000 0000 0000 0000 0000 0000 0000
012     BFC0 0000 0000 0000 0000 0000 0000 0000
013     BFC0 0000 0000 0000 0000 0000 0000 0000
014     BFC0 0000 0000 0000 0000 0000 0000 0000
015     BFC0 0000 0000 0000 0000 0000 0000 0000
016     BFC0 0000 0000 0000 0000 0000 0000 0000
017     BFC0 0000 0000 0000 0000 0000 0000 0000
018     BFC0 0000 0000 0000 0000 0000 0000 0000
019     BFC0 0000 0000 0000 0000 0000 0000 0000
01A     BFC0 0000 0000 0000 0000 0000 0000 0000
01B     BFC0 0000 0000 0000 0000 0000 0000 0000
01C     BFC0 0000 0000 0000 0000 0000 0000 0000
01D     BFC0 0000 0000 0000 0000 0000 0000 0000
01E     BFC0 0000 0000 0000 0000 0000 0000 0000
01F     BFC0 0000 0000 0000 0000 0000 0000 0000
```

Figure 3-28. AK,L Command Display for CYBER Model 990

CS Alter Command

The following alter command is associated with CS.

Enter CS (EK)

This command sends a master clear function to the selected element and causes data to be written into a specified location in CS. Data is entered from left to right. For example, if you enter 7F at address 834, only byte zero is altered.

Command format for CYBER models 810, 815, 825, 830, 835, 840, 845, 850, 855, 860, and CYBER 96X only:

```
EK,Pnum,My,adrs,byteno,data
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
y	CS memory select for CYBER models 840, 845, 850, 855, 860, and CYBER 96X only. 0 Enter primary. 1 Enter shadow (except 845 and 855). Omitted Enter both memories, if both memories exist.
adrs	The address where the data is to be written (hexadecimal, one to four characters). For CYBER 96X only, it determines lower or upper areas of CS. Address ranges are: Lower primary (M=0) 000_{16} to $7FF_{16}$ Upper primary (M=0) 800_{16} to FFF_{16} Lower shadow (M=1) 000_{16} to $7FF_{16}$ Upper shadow (M=1) 800_{16} to FFF_{16}
byteno	Byte number (0 through F) at which data entry will begin. If omitted, byte zero is assumed.
data	The data to be entered (1 to 32 hexadecimal digits).

Command format for CYBER 99X only:

```
EK,Pnum,L,adrs,byteno,data
```

where the following parameters apply: (num is the same as for CYBER 96X)

L	Logical enter. In logical format, bits entered into memory are arranged into logical groups instead of sequentially as in physical format. Physical format results when L is omitted.
adrs	The address where the data is to be written (hexadecimal, one to four characters).
byteno	The number (0 through 17) at which data entry will begin. If omitted, byte zero is assumed.
data	Logically formatted data to be entered.

CS Load Commands

The following load commands are associated with CS.

Load CS Program from MSL Device (CK)

This command sends a master clear function to the selected element and loads the specified program to CS, starting at the value of the fwa parameter. If only the first four characters of the program name are specified, the program loads from the CDA. If all seven characters are specified, the program loads from the MSL library. If the fwa parameter is not specified, the program is loaded at the address designated in the program tables.

Command format:

```
CK,Pnum,My,name,fwa

        or

CK,Pnum,My,name
```

where the following parameters apply:

num — CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.

y — CS memory select for CYBER models 840, 845, 850, 855, 860, and CYBER 96X only.

0	Load primary.
1	Load shadow (except 845 and 855).
Omitted	Load both memories, if both memories exist.

name — The name of the program. If first four characters are specified, program loads from CDA. If all seven characters are specified, program loads from the MSL library.

fwa — The first word address where the program is to be loaded (zero to four characters). If omitted, the entry in program tables is used. If no entry in the program tables exists, zero is assumed.

For CYBER 96X only, it determines lower or upper areas of CS. Address ranges are:

Lower primary (M=0)	000_{16} to $7FF_{16}$
Upper primary (M=0)	800_{16} to FFF_{16}
Lower shadow (M=1)	000_{16} to $7FF_{16}$
Upper shadow (M=1)	800_{16} to FFF_{16}

Load CS Program from MSL Device and Verify (VK)

This command loads the specified program into CS from the MSL device. Before each word is written to CS, a 32-bit checksum is generated. After CS is loaded, the words are read back from CS and another 32-bit checksum is generated. The two checksums are then compared and the system displays an error message if the checksums do not compare.

Command format:

```
VK,Pnum,My,name,fwa

      or

VK,Pnum,My,name
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
y	CS memory select for CYBER models 840, 845, 850, 855, 860, and CYBER 96X only. 0 — Load and verify primary. 1 — Load and verify shadow (except 845 and 855). Omitted — Load and verify both memories, if both memories exist.
name	The name of the program (one to seven alphanumeric characters).
fwa	The first word address where the program is to be loaded (zero to four characters). If omitted, the entry in program tables is used. If no entry in the program tables exists, zero is assumed. For CYBER 96X only, it determines lower or upper areas of CS. Address ranges are: Lower primary (M=0) — 000_{16} to $7FF_{16}$ Upper primary (M=0) — 800_{16} to FFF_{16} Lower shadow (M=1) — 000_{16} to $7FF_{16}$ Upper shadow (M=1) — 800_{16} to FFF_{16}

CS Execute Commands

The following execute commands are associated with the CS.

Deadstart CS (DK)

This command starts microcode execution, beginning at the CS address specified.

Command format:

```
DK,Pnum,My,adrs
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
y	CS memory select for CYBER models 840, 845, 850, 855, 860, and CYBER 96X only. 0 Deadstart primary. 1 Deadstart shadow (except 845 and 855). Omitted Deadstart memory selected by shadow select bit (bit 32 of DEC). Also, CS extended memory bit (bit 46 of DEC) for CYBER 96X only.
adrs	The starting address for microcode execution (zero to four characters). If omitted, execution starts at the last halt address.

Halt Microcode Execution (HK)

This command issues a stop processor function (00) to the CPU.

Command format:

```
HK,Pnum
```

where the following parameter applies:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.

Repetitive Deadstart CS (RK)

This command halts the CS processor, and restarts it at the address specified. A delay of approximately 1 microsecond for each delay count. This command remains in the keyboard buffer and executes until another keyboard entry is made.

Command format:

```
RK,Pnum,My,adrs,delay,C
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
y	CS memory select for CYBER models 840, 845, 850, 855, 860, and CYBER 96X only.
	0 — Repetitive deadstart primary.
	1 — Repetitive deadstart shadow (except 845 and 855).
	Omitted — Repetitive deadstart memory selected by shadow select bit (bit 32 of DEC). Also, CS extended memory bit (bit 46 of DEC) for model 96X only.
adrs	The CS address where execution is to begin (one to four digits).
delay	The delay count (one to three characters).
C	Master clear CMC for CYBER 99X only.

CS Library Command

The following library command is associated with CS.

Write CS to MSL Disk (*WK)

This command copies CS from fwa to lwa to the MSL disk device.

Command format:

```
*WK,Pnum,My,name,fwa,lwa+1
```

where the following parameters apply:

num — CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.

y — CS memory select for CYBER models 840, 845, 850, 855, 860, and CYBER 96X only.

0	Copy from primary.
1	Copy from shadow (except 845 and 855).
Omitted	Copy from memory selected by shadow select bit (bit 32 of DEC).

name — The program name (one to seven alphanumeric characters).

fwa — The first word address of the CS data (one to four characters).

lwa+1 — The last word address plus one of the CS data (one to four characters).

For CYBER 96X only, fwa/lwa determines lower or upper areas of CS. Address ranges are:

Lower primary (M=0)	000_{16} to $7FF_{16}$
Upper primary (M=0)	800_{16} to FFF_{16}
Lower shadow (M=1)	000_{16} to $7FF_{16}$
Upper shadow (M=1)	800_{16} to FFF_{16}

For CYBER 96X only, the upper address bit that determines lower or upper CS is not included in the load address that is written to disk.

Soft Control Memory Commands

The following commands perform operations with the soft memory.

Select Soft Control Memory Display (AS,BS)

This command displays a portion of the selected memory starting at the specified address.

Command format for CYBER 99X only:

```
AS,Pnum,mem,adrs  or  AS,Pnum,IBS,base,adrs
    or
BS,Pnum,mem,adrs  or  BS,Pnum,IBS,base,adrs
```

where the following parameters apply:

num — CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.

mem — Memory identifier (two to four characters).

IBS	Instruction Buffer Stack
IMAP	170 and 180 map
PM0	Page map set 0
PM1	Page map set 1
PM2	Page map set 2
PM3	Page map set 3
SM0	Segment map set 0
SM1	Segment map set 1
M2	Soft control M2
M3	Soft control M3
M4	Soft control M4
BDP	Business data processor
OCA	Operand cache
OCC	Fifo, prefetch, validity, tag
OCD	Data memory
RGA	A registers
RGX	X registers
LSU	Load/store unit
EPN	Error processing network
HF	History file
PMF	Maintenance register 22
Base	Data format for IBS:

O

Octal format

H

Hexadecimal format

Omitted

Hexadecimal

adrs — The relative first word address of data to be displayed. If omitted, display starts at address zero.

Examples of the AS command as it appears under the A or B header are shown in figures 3-29 through 3-41.

```
IMAP  CPO
000    0000 0000 0000 0000
001    0000 0000 0000 0000
002    0000 0000 0000 0000
003    0000 0000 0000 0000
004    0000 0000 0000 0000
005    0000 0000 0000 0000
006    0000 0000 0000 0000
007    0000 0000 0000 0000
008    0000 0000 0000 0000
009    0000 0000 0000 0000
00A    0000 0000 0000 0000
00B    0000 0000 0000 0000
00C    0000 0000 0000 0000
00D    0000 0000 0000 0000
00E    0000 0000 0000 0000
00F    0000 0000 0000 0000
010    0000 0000 0000 0000
011    0000 0000 0000 0000
012    0000 0000 0000 0000
013    0000 0000 0000 0000
014    0000 0000 0000 0000
015    0000 0000 0000 0000
016    0000 0000 0000 0000
017    0000 0000 0000 0000
018    0000 0000 0000 0000
019    0000 0000 0000 0000
01A    0000 0000 0000 0000
01B    0000 0000 0000 0000
01C    0000 0000 0000 0000
01D    0000 0000 0000 0000
01E    0000 0000 0000 0000
01F    0000 0000 0000 0000
```

Figure 3-29. AS, IMAP Command Display for CYBER 99X

```
PM0 CP0
000     F000 0000 0000 0000
001     F000 0000 0000 0000
002     F000 0000 0000 0000
003     F000 0000 0000 0000
004     F000 0000 0000 0000
005     F000 0000 0000 0000
006     F000 0000 0000 0000
007     F000 0000 0000 0000
008     F000 0000 0000 0000
009     F000 0000 0000 0000
00A     F000 0000 0000 0000
00B     F000 0000 0000 0000
00C     F000 0000 0000 0000
00D     F000 0000 0000 0000
00E     F000 0000 0000 0000
00F     F000 0000 0000 0000
010     F000 0000 0000 0000
011     F000 0000 0000 0000
012     F000 0000 0000 0000
013     F000 0000 0000 0000
014     F000 0000 0000 0000
015     F000 0000 0000 0000
016     F000 0000 0000 0000
017     F000 0000 0000 0000
018     F000 0000 0000 0000
019     F000 0000 0000 0000
01A     F000 0000 0000 0000
01B     F000 0000 0000 0000
01C     F000 0000 0000 0000
01D     F000 0000 0000 0000
01E     F000 0000 0000 0000
01F     F000 0000 0000 0000
```

Figure 3-30. AS, PM0 Command Display for CYBER 99X

```
SM0 CP0
000      F00F FF00 0000 0000
001      F00F FF00 0000 0000
002      F00F FF00 0000 0000
003      F00F FF00 0000 0000
004      F00F FF00 0000 0000
005      F00F FF00 0000 0000
006      F00F FF00 0000 0000
007      F00F FF00 0000 0000
008      F00F FF00 0000 0000
009      F00F FF00 0000 0000
00A      F00F FF00 0000 0000
00B      F00F FF00 0000 0000
00C      F00F FF00 0000 0000
00D      F00F FF00 0000 0000
00E      F00F FF00 0000 0000
00F      F00F FF00 0000 0000
010      F00F FF00 0000 0000
011      F00F FF00 0000 0000
012      F00F FF00 0000 0000
013      F00F FF00 0000 0000
014      F00F FF00 0000 0000
015      F00F FF00 0000 0000
016      F00F FF00 0000 0000
017      F00F FF00 0000 0000
018      F00F FF00 0000 0000
019      F00F FF00 0000 0000
01A      F00F FF00 0000 0000
01B      F00F FF00 0000 0000
01C      F00F FF00 0000 0000
01D      F00F FF00 0000 0000
01E      F00F FF00 0000 0000
01F      F000 0000 0000 0000
```

Figure 3-31. AS, SM0 Command Display for CYBER 99X

```
M2    CPO
000    0000 2000 0000 0000
001    0020 2000 0000 0000
002    0000 2000 1000 0000
003    0000 2000 0000 0000
004    0000 2000 0000 0000
005    0000 2000 0002 0000
006    0000 0000 0000 0000
007    0020 0000 0000 0000
008    0480 0200 8440 0900
009    8000 0000 0000 0000
00A    0480 0200 8240 0000
00B    8000 0000 0000 0000
00C    0000 0000 1000 0000
00D    0000 0000 1000 0000
00E    0000 2000 0000 0000
00F    0000 0000 1000 0000
010    0000 2000 0000 0000
011    0000 0000 1000 0000
012    0000 0000 1000 0000
013    0000 0000 0000 0000
014    0000 2000 0000 0000
015    0000 2000 0000 0000
016    0000 2000 1002 0000
017    0000 0000 1002 0000
018    0000 0000 1000 0000
019    0000 0000 1000 0000
01A    0180 0000 A080 1B00
01B    A780 0200 4000 1C00
01C    8000 2200 0000 0000
01D    0480 0200 0000 1E00
01E    2000 0000 0000 0000
01F    0004 0000 0000 0000
```

Figure 3-32. AS, M2 Command Display for CYBER 99X

```
BDP   CP0
000   0000 0000 0000 0000 0000 0000 0000 0000
001   0000 0000 2010 0000 0000 0000 0000 0000
002   0800 0120 2040 0300 0002 0040 0000 0300
003   0400 00A0 2040 0400 0002 0002 0000 0400
004   D354 5118 7000 0500 1250 0090 0200 0500
005   0001 1020 2040 0600 0000 0000 8200 0600
006   0001 0020 2040 0700 0010 0000 0000 0700
007   0000 0000 2040 0800 C004 1400 0000 0800
008   0000 0000 2040 0900 0000 0000 0200 0900
009   0000 0000 2040 0A00 0004 0000 0000 0A00
00A   0000 0000 0000 0A00 0000 0000 0200 0A00
00B   0008 0000 2040 0C00 0000 0002 0200 0C00
00C   13C0 0000 40A0 0E00 1000 0080 0200 0E00
00D   0000 0000 2040 0A00 0000 0000 0200 0A00
00E   0000 0000 0040 0F00 0000 0000 0200 0F00
00F   0000 0000 0040 1000 0000 0000 0200 1000
010   0000 0000 0040 1100 0000 0000 0200 1100
011   0000 0000 0040 1200 0000 0000 0200 1200
012   0000 0000 0040 1300 0000 0000 0200 1300
013   0000 0000 0040 0D00 0040 0000 0000 0D00
014   0000 0000 2040 0B00 0000 0000 0200 0B00
015   0000 0000 2010 0000 0000 0020 0000 0000
016   0000 0000 2010 0000 0000 0020 0000 0000
017   0000 0000 2010 0000 0000 0020 0000 0000
018   0000 0000 2010 0000 0000 0020 0000 0000
019   0000 0000 2010 0000 0000 0020 0000 0000
01A   0000 0000 2010 0000 0000 0020 0000 0000
01B   0000 0000 2010 0000 0000 0020 0000 0000
01C   0000 0000 2010 0000 0000 0020 0000 0000
01D   0000 0000 2010 0000 0000 0020 0000 0000
01E   0000 0000 2010 0000 0000 0020 0000 0000
01F   0000 0000 2010 0000 0020 0020 0000 0000
```

Figure 3-33. AS, BDP Command Display for CYBER 99X

```
OCA    CP0   COLUMN=00   FIFO= 8

SET    P   SVA-TAG          MDOCA   V   DATA

 0     0  0000 1FFF E000    1000    1   FFFF FFFF FFFF FFFF
                            1400    1   FFFF FFFF FFFF FFFF
                            1800    1   FFFF FFFF FFFF FFFF
                            1C00    1   FFFF FFFF FFFF FFFF

 1     0  0000 0000 0000    1100    1   FFFF FFFF FFFF FFFF
                            1500    1   FFFF FFFF FFFF FFFF
                            1900    1   FFFF FFFF FFFF FFFF
                            1D00    1   FFFF FFFF FFFF FFFF

 2     0  0000 1FFF E000    1200    1   FFFF FFFF FFFF FFFF
                            1600    1   FFFF FFFF FFFF FFFF
                            1A00    1   FFFF FFFF FFFF FFFF
                            1E00    1   FFFF FFFF FFFF FFFF

 3     0  0000 0000 0000    1300    1   FFFF FFFF FFFF FFFF
                            1700    1   FFFF FFFF FFFF FFFF
                            1B00    1   FFFF FFFF FFFF FFFF
                            1F00    1   FFFF FFFF FFFF FFFF
```

Figure 3-34. AS, OCA Command Display for CYBER 99X

```
M2    CP0
000    0000 4100 001F FFE0
001    0000 4100 001F FFE0
002    0000 4100 001F FFE0
003    0000 4100 001F FFE0
004    0000 4100 001F FFE0
005    0000 4100 001F FFE0
006    0000 4100 001F FFE0
007    0000 4100 001F FFE0
008    0000 4100 001F FFE0
009    0000 4100 001F FFE0
00A    0000 4100 001F FFE0
00B    0000 4100 001F FFE0
00C    0000 4100 001F FFE0
00D    0000 4100 001F FFE0
00E    0000 4100 001F FFE0
00F    0000 4100 001F FFE0
010    0000 4100 001F FFE0
011    0000 4100 001F FFE0
012    0000 4100 001F FFE0
013    0000 4100 001F FFE0
014    0000 4100 001F FFE0
015    0000 4100 001F FFE0
016    0000 4100 001F FFE0
017    0000 4100 001F FFE0
018    0000 4100 001F FFE0
019    0000 4100 001F FFE0
01A    0000 4100 001F FFE0
01B    0000 4100 001F FFE0
01C    0000 4100 001F FFE0
01D    0000 4100 001F FFE0
01E    0000 4100 001F FFE0
01F    0000 4100 001F FFE0
```

Figure 3-35. AS, OCC Command Display for CYBER 99X

```
RGA   CP0
000    0000 0000 0000 0000
001    0000 0000 0000 0001
002    0000 0000 0001 8000
003    0000 0000 0000 0000
004    0000 0000 0000 0001
005    0000 0000 0000 0000
006    0000 0000 0000 0001
007    0000 0000 0000 0000
008    0000 0000 0000 0000
009    0000 0000 0000 0000
00A    0000 0000 0000 0000
00B    0000 0000 0000 0000
00C    0000 0000 0000 0000
00D    0000 0000 0000 0000
00E    0000 0000 0000 0000
00F    0000 0000 0000 0000
```

Figure 3-36. AS, RGA Command Display for CYBER 99X

```
LSUP  CP0
000    0000 0000 0000 0000
001    0000 0004 0000 0000
002    0000 0004 0000 0000
003    0000 0004 0000 0000
004    0000 0004 0000 0000
005    0000 0004 0000 0000
006    0000 0004 0000 0000
007    0000 0004 0000 0000
008    0000 0004 0000 0000
009    0000 0004 0000 0000
00A    0000 0004 0000 0000
00B    0000 0004 0000 0000
00C    0000 0004 0000 0000
00D    0000 0004 0000 0000
00E    0000 0004 0000 0000
00F    0000 0004 0000 0000
010    0000 0004 0000 0000
011    0000 0004 0000 0000
012    0000 0004 0000 0000
013    0000 0004 0000 0000
014    0000 1A70 0000 0000
015    0000 0004 0000 0000
016    0000 0004 0000 0000
017    0000 0004 0000 0000
018    0000 0004 0000 0000
019    0000 0004 0000 0000
01A    0000 0004 0000 0000
01B    0000 0004 0000 0000
01C    0000 0004 0000 0000
01D    0000 0004 0000 0000
01E    0000 0004 0000 0000
01F    0000 0000 0000 0000
```

Figure 3-37. AS, LSU Command Display for CYBER 99X

```
LSUP  CP0
000   4000 4300 4000 4300
001   0003 C200 0003 C200
002   0008 0000 0008 0000
003   0008 0000 0008 0000
004   0008 0000 0008 0000
005   0008 0000 0008 0000
006   0008 0000 0008 0000
007   0008 0000 0008 0000
008   0008 0000 0008 0000
009   0008 0000 0008 0000
00A   0008 0000 0008 0000
00B   0008 0000 0008 0000
00C   0008 0000 0008 0000
00D   0008 0000 0008 0000
00E   0008 0000 0008 0000
00F   0008 0000 0008 0000
010   0008 0000 0008 0000
011   0008 0000 0008 0000
012   0008 0000 0008 0000
013   0008 0000 0008 0000
014   0008 0000 0008 0000
015   0008 0000 0008 0000
016   0008 0000 0008 0000
017   0008 0000 0008 0000
018   0008 0000 0008 0000
019   0008 0000 0008 0000
01A   0008 0000 0008 0000
01B   0008 0000 0008 0000
01C   0008 0000 0008 0000
01D   0008 0000 0008 0000
01E   0008 0000 0008 0000
01F   0000 0000 0000 0000
```

Figure 3-38. AS, EPN Command Display for CYBER 99X

```
HF     CP0
000    BAD0 BAD1 BAD2 BAD3 0000 0000 0000 0000
       BCD0 8000 0000 0000 0000 0000 0000 0000
001    BAD0 BAD1 BAD2 BAD3 0000 0000 0000 0000
       BCD0 8000 0000 0000 0000 0000 0000 0000
002    D000 0000 0000 0000 0000 0000 0000 0000
       8CD0 8000 0000 0000 0000 0000 0000 0000
003    0000 0000 0000 0000 0000 0000 0000 0000
       0000 0000 0000 0000 0000 0000 0000 0000
004    0000 0000 0000 0000 0000 0000 0000 0000
       0000 0000 0000 0000 0000 0000 0000 0000
005    0000 0000 0000 0000 0000 0000 0000 0000
       0008 0000 0000 0000 0000 0000 0000 0000
006    0000 0000 0000 0000 0000 0000 0000 0000
       0000 0000 0000 0000 0000 0000 0000 0000
007    0000 0000 0000 0000 0000 0000 0000 0000
       0000 0000 0000 1000 0000 0000 0200 1000
008    0000 0000 0000 1100 0000 0000 0200 1100
       0000 0000 0000 1200 0000 0000 0200 1200
009    0000 0000 0000 1300 0000 0000 0200 1300
       0000 0000 0000 0D00 0040 0000 0000 0D00
00A    0000 0000 0000 0B00 0000 0000 0200 0B00
       0000 0000 0000 0000 0000 0020 0000 0000
00B    0000 0000 0000 0000 0000 0020 0000 0000
       0000 0000 0000 0000 0000 0020 0000 0000
00C    0000 0000 0000 0000 0000 0020 0000 0000
       0000 0000 0000 0000 0000 0020 0000 0000
00D    0000 0000 0000 0000 0000 0020 0000 0000
       0000 0000 0000 0000 0000 0020 0000 0000
00E    0000 0000 0000 0000 0000 0020 0000 0000
       0000 0000 0000 0000 0000 0020 0000 0000
00F    0000 0000 0000 0000 0000 0020 0000 0000
       0000 0000 0000 0000 0000 0020 0000 0000
```

Figure 3-39. AS, HF Command Display for CYBER 99X

```
PMF    CP0
000    00 00 00 00 00 00 00 00
008    0F 0F 1F 1F 1F 1F 1F 1F
010    00 00 00 00 00 00 00 00
018    00 00 00 00 00 00 00 00
020    00 00 00 00 00 00 00 00
028    00 00 00 00 00 00 00 00
```

Figure 3-40. AS, PMF Command Display for CYBER 99X

```
IBS0   CP0
000    34400 02203 00000 02203
001    00000 00000 00000 00000
002    52000 00000 00000 00000
003    00000 00000 00000 00000
004    52400 00000 00000 00000
005    00000 00000 00000 00000
006    53000 00000 00000 00000
007    00000 00000 00000 00000
008    53400 00000 00000 00000
009    00000 00000 00000 00000
00A    54000 00000 00000 00000
00B    00000 00000 00000 00000
0BC    54400 00000 00000 00000
00D    00000 00000 00000 00000
00E    55000 00000 00000 00000
00F    43600 00000 00000 00000
010    34400 02203 00000 02203
011    00000 00000 00000 00000
012    40600 00000 41400 00000
013    00000 00000 00000 00000
014    00000 00000 00000 00000
015    00000 00000 00000 00000
016    00000 00000 00000 00000
017    00000 00000 00000 00000
018    00000 00000 00000 00000
019    00000 00000 00000 00000
01A    00000 00000 00000 00000
01B    00000 00000 00000 00000
01C    00000 00000 00000 00000
01D    00000 00000 00000 00000
01E    00000 00000 00000 00000
01F    00000 00000 00000 00000
```

Figure 3-41. AS, IBS Command Display for CYBER 99X

Load Soft Control Memory from MSL (CS)

This command sends a master clear function to the system processor and loads the named program to the selected soft memory.

Command format for models 840, 845, 850, 855, 860, and 96X only:

```
CS,Pnum,name,t,fwa
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
name	Program name (one to seven alphanumeric characters).
t	MCH type code of soft memory to be loaded (three to seven). Type code 7 is used for register file access.
fwa	First word address where program is to be loaded. If omitted, program is loaded starting at assembly address.

The following command loads the named program to the selected memory starting at the specified address.

Command format for CYBER 99X only:

```
CS,Pnum,mem,adrs
```

where the following parameters apply: (num and name are the same as for CYBER 96X)

mem — Memory identifier (one to seven alphanumeric characters).

IBS	Instruction buffer stack
IMAP	170 and 180 map
PM0	Page map set 0
PM1	Page map set 1
PM2	Page map set 2
PM3	Page map set 3
SM0	Segment map set 0
SM1	Segment map set 1
M2	Soft control M2
M3	Soft control M3
M4	Soft control M4
BDP	Business data processor
OCA	Operand cache
OCC	Fifo, prefetch, validity, tag
OCD	Data memory
RGA	A registers
RGX	X registers
LSU	Load/store unit
EPN	Error processing network
HF	History file
PMF	Maintenance register 22

adrs — Relative hexadecimal address where program is to be loaded. If omitted, absolute MAC address in program load table is used.

Alter Soft Control Memory (ES)

This command sends a master clear to the processor and alters the selected location.

Command format for models 840, 845, 850, 855, 860 and CYBER 96X only:

```
ES,Pnum,t,adrs,data
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
t	MCH type code of soft memory to be altered (three to seven). Type code 7 is used for register file access.
adrs	Address to be altered.
data	Hexadecimal data to be inserted (one to eight digits for soft memory, or 1 to 16 digits for register file).

The following command sends data into a byte or bytes of the selected memory.

Command format for CYBER 99X only:

```
ES,Pnum,mem,adrs,byte,data
```

where the following parameters apply: (num is the same as for CYBER 96X)

mem — Memory identifier (two to four characters). If BP3 is specified, all bytes of specified address which are not entered will be zeros.

IBS	Instruction buffer stack
IMAP	170 and 180 map
PM0	Page map set 0
PM1	Page map set 1
PM2	Page map set 2
PM3	Page map set 3
SM0	Segment map set 0
SM1	Segment map set 1
M2	Soft control M2
M3	Soft control M3
M4	Soft control M4
BDP	Business data processor
BP3	P3 part of BDP
OCA	Operand cache
OCC	Fifo, prefetch, validity, tag
OCD	Data memory
RGA	A registers
RGX	X registers
LSU	Load/store unit
EPN	Error processing network
HF	History file
PMF	Maintenance register 22

adrs	The relative address to be altered. Omitted for PMF.
byte	The starting hexadecimal byte number of data to be entered. If omitted, byte 0 is selected. Each byte is eight bits. Byte 0 reflects the highest order eight bits. Use the following byte numbers.

BDP	0 through F
HF	0 through 1F
PMF	0 through 2F
All other memory	0 through 7

data	Hexadecimal data to be entered starting at the specified byte number and proceeding to the lower order bits. If one character of data is entered, it is right-justified with a padded zero in the specified byte.

Write Soft Control Memory to MSL Disk (*WS)

This command copies the specified memory from fwa to lwa to the MSL disk as a file under the specified program name.

Command format for CYBER 99X only:

```
*WS,Pnum,name,mem,fwa,lwa+1
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
name	Program name (one to seven alphanumeric characters).
mem	Memory identifier (two to four characters). BP3 are write only memories and cannot be read, then written to disk.

IBS	Instruction buffer stack
IMAP	170 and 180 map
PM0	Page map set 0
PM1	Page map set 1
PM2	Page map set 2
PM3	Page map set 3
SM0	Segment map set 0
SM1	Segment map set 1
M2	Soft control M2
M3	Soft control M3
M4	Soft control M4
BDP	Business data processor
BP3	P3 part of BDP
OCA	Operand cache
OCC	Fifo, prefetch, validity, tag
OCD	Data memory
RGA	A registers
RGX	X registers
LSU	Load/store unit
EPN	Error processing network
HF	History file
PMF	Maintenance register 22

fwa	The relative first word address from which program starts to be written to disk.
lwa+1	The relative last word address plus one at which program to be written to disk will end.

Special Memory Commands

The following commands are associated with special memory.

Special Memory Display Commands

The following display commands are associated with special memory.

Select Special Memory Display (AR,BR)

The following command displays the contents of the CIR or EIT memory.

Command format for CYBER 99X only:

```
AR,Pnum,mem

    or

BR,Pnum,mem
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
mem	Memory identifier (two or three characters). CIR Current instruction register EIT Error information table

An example of an AR COMMAND display as it appears under the A or B header is shown in figure 3-42.

```
EIT  CPO
000  0000 0000 0000 F200
001  0000 0000 0101 F210
002  0000 0000 0202 F220
003  0000 0000 0303 F230
004  0000 0000 0404 F240
005  0000 0000 0505 F250
006  0000 0000 0606 F260
007  0000 0000 0707 F270
008  0000 0000 0808 F280
009  0000 0000 0909 F290
00A  0000 0000 0A0A F2A0
00B  0000 0000 0B0B F2B0
00C  0000 0000 0C0C F2C0
00D  0000 0000 0D0D F2D0
00E  0000 0000 0E0E F2E0
00F  0000 0000 0F0F F2F0
```

Figure 3-42. AR, EIT Command Display for CYBER 99X

The following command displays the contents of an A or X scratch register.

Command format for CYBER 99X only:

```
AR,Pnum,SCR,name,reg

        or

BR,Pnum,SCR,name,reg
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
name	Name of display register.
	A A register
	X X register
reg	The register number to be displayed.

Control Word Display (AT,BT)

This command displays the contents of the control word memory in logical or physical format. In logical format, bits read from memory are arranged in logical groups for display purposes instead of sequentially as in physical format. Logical format is arranged logically to match the microcode remedy listing.

Command format for CYBER 99X only:

```
AT,Pnum,disp,adrs

        or

BT,Pnum,disp,adrs
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
disp	Type of display. L Logical Omitted Physical
adrs	The first word address of the data from which to start the display.

Examples of physical and logical formats of the AT COMMAND displays as they appear under the A or B header are shown in figures 3-43 and 3-44.

```
AT  CP0  000
000   0000 0800 0000 2000 0000 2000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
001   0000 0800 0000 2000 0000 2000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
002   0000 0800 0000 2000 0000 2000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
003   0000 0800 0000 2000 0000 2000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
004   38A0 2A00 000B 2800 C000 2000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
005   38A0 4A00 0008 E800 C000 2000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
006   38A0 2AD0 000B 2800 8000 3000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
007   38A0 4AD0 0008 E800 8000 2000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 1000
008   0000 0800 0000 2000 0008 2000 0000 0000
      1000 0000 0000 0000 0000 0000 0100 0000
009   0000 0800 0000 2000 0000 2000 0000 0000
      0000 0000 0000 0000 0000 0000 0001 0000
00A   0000 0800 0000 2000 0000 2000 0000 0000
      0000 0000 0000 0000 0000 0000 0068 0000
00B   38A0 4A00 0008 E800 9000 2000 0000 0A00
      0000 0000 0000 0000 0000 0000 0000 1000
00C   0000 0800 0000 2000 0000 2000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
00D   38A0 2AD0 000B 2800 A000 3000 0000 0500
      0000 0AD0 0000 0000 0000 0000 0000 0000
00E   BF10 AA00 2003 2400 8010 2000 A400 0000
      1000 0000 0200 0000 0042 0000 0100 0000
00F   0000 0800 0000 2000 0000 2000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
```

Figure 3-43. AT Command Display for CYBER 99X

```
ATL  CP0  000
000   0000 0001 0C00 0000 0000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
001   0000 0001 0C00 0000 C000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
002   0000 0001 0C00 0000 0040 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
003   0000 0001 0C00 0000 0000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
004   0000 0008 3800 0000 0008 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
005   0000 0009 0800 0000 0000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
006   0000 0001 2C00 0000 0028 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
007   0000 0001 0C00 0000 8013 FC00 0000 0000
      0000 0000 0000 1000 0000 0000 0000 0000
008   0000 0001 0C00 0000 0040 0000 0000 0000
      0000 0000 0000 0004 0000 0000 0000 0000
009   0000 0001 0C00 0000 0000 0000 0000 0000
      0000 0000 0000 0500 0000 0000 0000 0000
00A   38A0 2018 872F 00AC 0040 0000 0000 0000
      0000 0000 0000 0004 0000 0000 0000 0000
00B   38A0 2019 B72D 00AC 0008 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
00C   0000 0081 0C00 0002 8043 8401 0000 1000
      0000 0000 0000 1500 0000 0000 0000 0000
00D   38AC 4001 0CED 0150 0000 0000 0000 0000
      0000 A000 0000 0000 0000 0000 0000 0000
00E   38A0 2081 0F2D 00AD 0000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
00F   0000 0001 0C00 0000 0100 0000 0000 0040
      2000 0000 0000 0000 0000 0000 0000 0000
```

Figure 3-44. AT,L Command Display for CYBER 99X

Move Capture Buffer to CM for Display (MX)

This command transfers the 512-word capture buffer to CM beginning at the address specified. The 512-word block of CM is set to all Fs prior to the transfer.

If you enter K for the address portion of the command, the capture buffer is cleared by issuing a zero pattern function.

Command format for CYBER 99X only:

```
MX,adrs

 or

MX,K
```

where the parameters are as follows:

adrs	The hexadecimal CM address where data is to be written.
K	Causes the capture buffer to be cleared.

Control Word Alter Command

The following alter command is associated with control word.

Enter Control Word (ET)

This command enters data into a byte or bytes of control word memory in logical or physical format. In logical format, bits are arranged in logical groups instead of sequentially as in physical format.

Command format (for CYBER 99X only):

```
ET,Pnum,L,adrs,byte,data
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
L	Logically formatted data for model 990 only.
adrs	Relative hexadecimal memory address to be altered.
byte	The starting hexadecimal byte number of data to be entered. If omitted, byte 0 is selected. Byte 0 corresponds to the highest order eight bits. Valid byte numbers are 0 through 1F.
data	Hexadecimal data to be entered starting at the specified byte number and proceeding to the lower order bits. If one character of data is entered, it is right-justified with a padded zero in the specified byte.

Control Word Load Command

The following load command is associated with the control word.

Load Control Word Program from MSL (CT,VT)

This command loads the selected program into control word memory from MSL.

The VT command also verifies the loaded data. Before each word is written to control word, a 32-bit checksum is generated. After control word is loaded, the words are read back to the monitor and another checksum is generated. If the checksums differ, an error message is issued.

Command format (for CYBER 99X only):

```
CT,Pnum,name,adrs

        or

VT,Pnum,name,adrs
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
name	The program name (one to seven characters).
adrs	The relative hexadecimal address where program is to be loaded. If omitted, program is loaded at absolute MAC address in program load table.

Control Word Library Command

The following library command is associated with the control word.

Write Control Word to MSL Disk (*WT)

This command copies control word memory from fwa to lwa to the MSL disk as a file under the specified program name.

Command format (for CYBER 99X only):

```
*WT,Pnum,name,fwa,lwa+1
```

where the following parameters apply:

num	CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.
name	The program name (one to seven characters).
fwa	The relative first word address in the control word from which program starts to be written to disk.
lwa+1	The relative last word address plus one in the control word at which program to be written to disk will end.

Maintenance Channel Commands

The following commands are associated with the maintenance channel (MCH).

MCH Display Command

The following display command is associated with the MCH.

Select Maintenance Register Display (AR, BR)

Depending on the command format and the parameters entered, the R display shows one of three levels of register detail.

Level 1 command format:

```
AR

  or

BR
```

This format causes a nondynamic display of the status summary registers of the PP, CM, and CP system elements. The system displays a descriptive line for each failure bit or informative bit set and for each group of bits (for example, an address) regardless of the set condition. The line contains the bit number, name, and value.

Level 2 command format:

```
AR,el,S

  or

BR,el,S
```

where the following parameters apply:

el Selection of mainframe element.

Numeric

0	IOU
1	Memory
2	CPU 0

Alphabetic

C	CEM
I	IOU
M	Memory
P	Processor
Pnum	CPU num selected. If num is omitted, the default is assumed.

S Selection of soft register displays (CPU register display only).[5] If omitted, hard registers are displayed.

5. Microcode must be running to display soft registers.

This format displays all the registers of the selected element. The display format is: register number, four groups of four hexadecimal digits of dynamic date, and register name. If you select the CPU REGISTER display, an additional parameter allows you to display of the soft registers. If the parameter is omitted, the system displays only the actual CPU register information.

Level 3 command format:

```
AR,el,reg

  or

BR,el,reg
```

where the following parameters apply:

el	Mainframe element selection.
	Numeric
	0 IOU 1 Memory 2 CPU 0
	Alphabetic
	C CEM I IOU M Memory P Processor Pnum CPU num selected. If num is omitted, the default is assumed.
reg	Register number to be displayed (one or two hexadecimal digits). Default to level 2 if reg is omitted.

This format displays the nondynamic contents of a specified hard register. The system displays a descriptive line for each failure bit or informative bit that is set as well as for each group of bits, (for example, an address) regardless of the set condition.

NOTE

If CS is halted, all zeros are returned for a read of the soft registers. The register display shows all zeros. If CS is hung, no data is returned for a read of the soft registers and the display is blank.

Examples of the AR COMMAND display are shown in figures 3-45, 3-46, and 3-47.

```
                I2    STATUS SUMMARY = 14

BIT 59 - SUMMARY STATUS              = 1
BIT 61 - UNCORRECTED ERROR           = 1

                M5    STATUS SUMMARY = 04

BIT 61 - UNCORRECTED ERR             = 1

                P5    CP0 STATUS SUMMARY = 2C

BIT 58 - C180 MONITOR MODE           = 1
BIT 60 - PROCESSOR HALTED            = 1
BIT 61 - UNCORRECTABLE ERR           = 1

                P5    CP1 STATUS SUMMARY = 2C

BIT 58 - C180 MONITOR MODE           = 1
BIT 60 - PROCESSOR HALTED            = 1
BIT 61 - UNCORRECTABLE ERR           = 1
```

Figure 3-45. AR Level 1 Command Display

```
P5 CPO REGISTER DATA

13    0000 0000 0000 C000    VIRT MACH CAPABLTY
40    0000 0000 FFFF FFFF    PROGRAM ADDRESS
41    0000 0000 0000 0000    MON PROCESS STATE
42    0000 0000 0000 984E    MONITOR CONDITION
43    0000 0000 0000 4039    USER CONDITION
44    0000 0000 0000 0000    UNTRANSLATABLE PTR
45    0000 0000 0000 0000    SEGMENT TBL LENGTH
46    0000 0000 0000 0000    SEGMENT TBL ADRS
47    0000 0000 0000 0000    BASE CONSTANT
48    0000 0000 0000 1000    PAGE TABLE ADDRESS
49    0000 0000 0000 0000    PAGE TABLE LENGTH
4A    0000 0000 0000 0000    PAGE SIZE MASK
50    0000 0000 0000 0000    MODEL DPNDNT FLAGS
51    0000 0000 0000 0000    MODEL DPNDNT WORD
60    0000 0000 0000 0000    JOB PROCESS STATE
61    0000 0000 0000 0000    JOB PROCESS STATE
62    0000 0000 FFFF FFFF    SYS INTERVAL TIMER
63    0000 0000 0000 0000    KEYPOINT BUFFER POINTER
C0    0000 0000 0000 0000    TRAP ENABLE
C4    0000 0000 0000 0000    TRAP POINTER
C5    0000 0000 0000 0000    DEBUG LIST POINTER
C6    0000 0000 0000 0000    KEYPOINT MASK
C9    0000 0000 FFFF FFFF    PROCESS INTRVL TME
E0    0000 0000 0000 0000    CRITICAL FRAME FLG
E2    0000 0000 0000 0000    ON CONDITION FLAG
E4    0000 0000 0000 0000    DEBUG INDEX
E5    0000 0000 0000 0000    DEBUG MASK
E6    0000 0000 0000 FE00    USER MASK
```

Figure 3-46. AR Level 2 Command Display

```
                    P5 CPO REGISTER 10
                    ELEMENT ID
                    0000 0000 0040 0102

BIT 32-39 ELEMENT NUMBER        = 00
BIT 40-47 MODEL NUMBER          = 40
BIT 48-63 CPU SERIAL NUMBER     = 0102
```

Figure 3-47. AR Level 3 Command Display

MCH Alter Command

The following alter command is associated with the MCH.

Enter Maintenance Register Data (ER)

This command enters data into the selected IOU, memory, or processor maintenance register.

The ER command performs the following steps.

1. Reads the contents of the selected register.
2. Logically ANDs the register contents with the mask.
3. Logically ORs the result of step 2 with the data parameter.
4. Enters the result of step 3 into the register.

Omitting the mask causes the 64 bits of data in the register to be replaced with the data parameter supplied in the command.

Command format:

```
ER,el,reg,data,mask
```

where the following parameters apply:

el	Mainframe element selection.	
	0 or I	IOU
	1 or M	Memory
	2	CPU 0
	Pnum	CPU num selected. If omitted, the default CPU is assumed.
reg	The register to be altered (one or two digits).	
data	1 to 16 hexadecimal characters, right-justified.	
mask	1 to 16 hexadecimal character mask. If omitted, 0 is assumed.	

MCH Execute Commands

The following execute commands are associated with the MCH.

Master Clear Via MCH (CX)

This command causes a master clear function to be executed on the selected MCH access.

Command format:

```
CX,el
```

where the following parameters apply:

el Mainframe element selection.

0 or I	IOU
1 or M	Memory
2	CPU 0
Pnum	CPU num selected. If omitted, the default CPU is assumed.

Clear Error Via MCH (CE)

This command causes a clear error to be executed on the selected MCH access.

Command format:

```
CE,el
```

where the following parameters apply:

el Mainframe element selection.

0 or I	IOU
1 or M	Memory
2	CPU0
Pnum	CPU num selected. If omitted, the default CPU is assumed.

Command Buffer Commands

The following commands pertain to command buffers. These commands are available with the disk library only.

Through entering one command, command buffers allow the saving and easy execution of a string of CMSE and/or program commands. The CMSE and/or program commands are necessary to execute a particular diagnostic sequence.

Command Buffer Display Commands

The following display commands are associated with command buffers.

Display Command Buffer (AE, BE)

This command places a selected command buffer into the command buffer editing utility (CBU) PP for display and modification. The IN, DE, GO, and *WB commands affect the command buffer placed in the CBU PP. If CBU was not loaded into a PP by a previous AE command, the AE command also causes CBU to be loaded into a PP before the command buffer to be edited is loaded. If no name was given in the first AE command, CBU attempts to recover the command buffer that was originally loaded in the PP memory.

Command format:

```
AE,name,num

    or

BE,name,num
```

where the following parameters apply:

name	The name of the command buffer to be displayed (one to seven alphanumeric characters). If omitted, the system displays the command buffer currently in the CBU PP.
num	The number of the first command to be displayed. If omitted, the display starts with the first command.

Two examples of the AE COMMAND display as they appear below the A header are shown in figures 3-48 and 3-49.

```
EDIT    COMMAND BUFFER
0000    *MCODE
0001    CK U2JUN20
0002    EK,835,A285 2000 0014 8000 0000
        01E0 0004 002E
0003    EK,852,E287 7000 0014 8000 0000
        01E0 0004 002E
0004    EK,853,A283 C000 0014 8000 0000
        01E0 0004 002E
0005    EK,871,E287 7000 0014 8000 0000
        05E0 0004 002E
0006    EK,873,E287 7000 0014 8000 0000
        05E0 0004 002E
0007    EK,875,E287 7000 0014 8000 0000
        05E0 0004 002E
0010    EK,699,A269 E612 00E1 AE78 0000
        05E0 0004 002E
0011    EK,366,8200 00DA FB1B 2902 4238
        2000 45ED 0005
0012    EK,382,A038 801A F21E 0981 0238
        0000 07BC 8006
0013    EK 38C 8200 00DA F21E 0981 0238
        0000 07BC 8006
0014    EK,388,A039 601C F21D 2983 0238
        0000 00D9 8381
```

Figure 3-48. AE Command Display (1)

```
EDIT    COMMAND BUFFER
0000    *DMP
0001    DP10
0002    LT10,DMP,,1305
```

Figure 3-49. AE Command Display (2)

Display Command Buffer Name Table (AG, BG)

This command displays up to 200 command buffer names beginning with the first command buffer name in the library or beginning at the index.

Command format:

```
AG,indx

  or

AG

  or

BG,indx

  or

BG
```

where the following parameter applies:

indx   Offset from the first name in the table (one to four octal digits). If omitted, the display begins at the first name.

An example of the AG COMMAND display as it appears below the A header is shown in figure 3-50.

```
CMT       CMT2      LTREK     IM2
LTRAP     LRCT3C    EXN       EXNCT3
CT3PK     LTREKC    XNCT3C    LNOSC
LNOSC1    LNSALX    PATCH     LXCALX
CDMP      464SP     JA10FX    MEMOVE
LOPHCS    LOOK      LFCT2P    LTREK1
LTREK1C   HCSCAT    LTREKP    LTREKPC
LNOSCU1   LCT3C     LCT3      FEB28FX
LOPHCSC   LHCS1     HCSINFO   LOPHCS1
LHCSCIP   LHCBUF    HCSIDEA   TREKFIX
JFS1      JANIDEA   LHCSCA    EXNCU1
ASS       SNAP      SNAPA     SNAPB
KLEAN     LHCSC11   LHCSC2    LHCSC4
LHCSC1    LTEST     ASS3      LHCST
MONITOR   LHCST1    BKUP      IOUTEST
DMP       CALL      END
```

Figure 3-50. AG Command Display

Command Buffer Alter Commands

The following alter commands are associated with command buffers.

Insert Command in Command Buffer (IN)

This command inserts a command following the specified command. Continuous entries may be made by using the right space key (forward key), which increments the entry number and positions the keyboard line for the command entry. Refer to Keyboard Usage, section 3.

Command format:

```
IN,num,cmnd
```

where the following parameters apply:

num	The number of the command preceding the location where the new command is to be inserted. The commands are numbered in octal.
cmnd	The command to be inserted.

Delete Command in Command Buffer (DE)

This commands deletes or replaces the specified command.

Command format:

```
DE,num

  or

DE,num,cmnd
```

where the following parameters apply:

num	The number of the command to be deleted or replaced.
cmnd	The command to replace the deleted command.

Command Buffer Execute Commands

The following execute commands are associated with command buffers.

Execute Command Buffer (GO)

This command causes the commands in the command buffer to be executed sequentially. If the command is entered without a name, the next command in a halted command buffer is executed. If an asterisk (*) is used, the command buffer currently residing on the edit track is executed. Command buffer execution may be halted by pressing the **Erase** key.

Command format:

```
GO,name

  or

GO,*
```

where the following parameter applies:

name	The name of the command buffer, one to seven alphanumeric characters.

Terminate Command Buffer (TB)

This command terminates the current active command buffer.

Command format:

```
TB
```

These commands terminate all active or waiting command buffers.

Command Format:

```
TB*

  or

TB,*
```

Temporarily Halt Command Buffer Sequence (SQ)

This command halts the execution of the command buffer until the value of adrs is greater than or equal to the value of tshld.

If the sequence address is unknown or does not exist, use the alternate format. The alternate format halts execution until the number of passes through the monitor, specified by monpass, have been completed.

Command format:

```
SQ,base,adrs,tshld,mem
```

where the following parameters apply:

base — Indicates the command buffer format as follows:

O	Octal
H	Hexadecimal
Omitted	Hexadecimal

adrs — The address to be monitored for the command buffer status (one to eight characters).

tshld — The threshold to be monitored for command buffer status (16 bits maximum).

mem — The type of memory to be monitored. If omitted, 16 is assumed. A slash (/) causes automatic PP to be substituted.

00-11	PPs
16	CM
20-31	PPs (C0 through C11 for AT478/AT481 CIO PPs)

Alternate command format:

```
SQ,Base,adrs,count,17
```

where the following parameters apply:

base — Same as above.

adrs — The address must be zero.

count — Monitor pass count (1 through 7777_8)

Display Comment

A command buffer entry with a blank in the first column halts command buffer execution until you enter a GO command. The system displays the remainder of the entry until this occurs.

Command format:

```
(blank in first column) comment
```

Send Space Command (SP)

This command is translated as a space bar entry from the keyboard for use as a program execution command for command buffers.

Command format:

```
SP
```

Return Jump To Command Buffer (RJ)

This command causes the current command buffer sequence to be interrupted. The alternate command buffer specified by the name parameter is executed, beginning with the first command. When the alternate command buffer is completed, control is returned to the original command buffer at the command after the RJ command. Primary command buffers can call secondary command buffers, and secondary command buffers can return jump to third-level command buffers, but third-level command buffers cannot return jump to other command buffers.

Command format:

```
RJ,name
```

where the following parameter applies:

name	Name of command buffer to jump to (one to seven characters).

Drop Command Buffer Editing Utility (KE)

This command causes CMSE to drop CBU and make its PP available for other purposes.

Command format:

```
KE
```

Display Message on A or B Screen (AM,BM)

This command displays a message on the A or B screen on the specified line. If the screen is currently displaying messages from AM or BM commands, the message is added to the screen on the specified line overwriting any message on that line. If the screen is displaying any other display, the screen is cleared and the message is displayed on the specified line. This command is used in command buffers to display important messages related to the execution of a command buffer.

Command format:

```
AM,linenum,msg

  or

BM,linenum,msg
```

where the following parameters apply:

linenum	Hexadecimal line number 0 through 13 on screen where message is to be displayed. This limits the display to the size of a 721 terminal screen.
msg	The message to be displayed.

Command Buffer Comment Line (.)

Any command buffer command which begins with a period is treated by CMSE as a comment line and is not examined as a command for execution. This provides a means of putting comment lines into command buffers without forcing CMSE to look at more than one character to decide if the line is a command or a comment line. A comment line is not executed when the command buffer is executed but shows up on command buffer displays.

Command format:

```
. (period)
```

Command Buffer Library Commands

The following library commands are associated with command buffers. Section 4 contains a description of the procedures required to manipulate command buffers and programs on the disk library.

Write Command Buffer to MSL Disk (*WB)

This command causes the contents of the edit track to be written to the command buffer area of the disk.

Command format:

```
*WB
```

Delete Command Buffer from Disk Library (*DB)

This command deletes a specified command buffer from the disk library. If the form *DB * is used, all command buffers are deleted. The *DB * format should be used to initialize the command buffer area after MSL is loaded to the disk.

Command format:

```
*DB,name

   or

*DB *
```

where the following parameters apply:

name	The name of the command buffer to be deleted (one to seven alphanumeric characters).

CAUTION

The *DB * form of this command deletes all command buffers from the connected disk.

CMSE Commands

The following commands are associated with CMSE.

CMSE Disk Library Commands

The following commands pertain to the CMSE disk library.

Delete Program From MSL Disk (*DP)

This command deletes a specified program from the disk library. A special form of this command purges the dayfile.

Command format:

```
*DP,name
```

where the following parameters apply:

name	The name of the program to be deleted (one to seven alphanumeric characters).

Delete Dayfile (*DP,*DF)

This command purges the dayfile from the disk.

Command format:

```
*DP,*DF
```

Display Program Name Table (AF, BF)

This command displays up to 200 program names, beginning with the first program in the library (default), or beginning at the index.

Command format:

```
AF,indx

  or

BF,indx
```

where the following parameter applies:

indx	Offset from the first name in the table (one to four octal digits). If omitted, the display begins with the first name.

An example of the AF COMMAND display as it appears below the A header is shown in figure 3-51.

```
PROGRAM NAMES        INDEX 00000
   CEY       CET       CEX       CE1
   DDS       MSB       DSB       D01
   D02       D03       D04       D05
   D06       D07       BIF       MEP
   MEP000    MEP001    MEP002    MEP003
   MEP004    MEP005    MEP006    MEP007
   MEP010    MEP011    MEP012    MEP013
   MEP014    MEP015    MEP016    MEP017
   MEP020    MEP021    MEP022    MEP023
   MEP024    MEP025    MEP026    MEP027
   MEP030    MEP031    MEP032    MEP033
   MEP034    MEP035    MEP036    MEP037
   MEP040    MEP041    MEP042    MEP043
   MEP044    MEP045    MEP046    MEP047
   MEP050    MEP051    MEP052    MEP053
   MEP054    MEP055    MEP056    MEP057
   DMP       BCS       BCF       FMD
   FIRM66X   TDX       DSP       RMD
   RCT1000   CDT       9XI       9PT
   9PD       PCD       9CT       U2APRO2
   EI        DXX       DDD       ODDD
   CLK       PSP       PMM       QLT2
   CMT2      CMT20V0   CM01      CM02
   CM03      CM04      CM05      CM06
   CM07      CM08      CM09      CM10
   CM11      CM12      CM13      CM14
   CM15      CM16      CM17      CM18
   CM19      CM20      CM21      CM22
   CM23      CM24      CM25      CM26
   M2ET      CM30      EDS2      EXC
   9AX       9AY       CPR       CM6
   CT3       CMC       EJP       3D1
```

Figure 3-51. AF Command Display

Display Dayfile (AY, BY)

This command displays the contents of the dayfile beginning with the entry number. If the indx parameter is omitted, the display begins with the first entry.

```
AY,indx

  or

BY,indx
```

where the following parameter applies:

indx Entry number (one to four octal digits). Default is zero.

An example of the AY COMMAND display as it appears below the A header is shown in figure 3-52.

```
DAYFILE INDEX 0000
MONITOR SOFT ERROR   P = 0000B00800001770   MCR = 0002

     PFS0 = 4000000004000000              PFS1 = 0000

     PFS2 = 0000000000000000              PFS3 = 0000

     PFS4 = 0000000000000000              PFS5 = 0000

     PFS6 = 0000000000000000              PFS7 = 0000

     PFS8 = 0000000000000000              FPS9 = 0000

MONITOR SOFT ERROR P = 0000B00800001C58 MCR = 0002

     PFS0 = 16MONITOR SOFT ERROR P = 0000B00B008000013C4 MCR = 042

     PFS0 = 16MONITOR SOFT ERROR P = 0000B00800001CA8 MCR = 0002

     PFS0 = 16MONITOR SOFT ERROR P = 0000B008000013C4 MCR = 0042

     PFS0 = 16MONITOR SOFT ERROR P = 0000B008000013C4 MCR = 0042
```

Figure 3-52. AY Command Display

Enter Dayfile Command (ED)

This command adds the message or creates and adds the message to a file named *DF. The message appears on either the AY or BY display following the mnemonic KYBD, which indicates that it originated from the keyboard. To purge the dayfile, use the command *DP,*DF.

Command format:

```
ED,msg
```

where the following parameter applies:

msg	The message to be entered (up to 64 alphanumeric characters).

CMSE Utility Commands

The following commands pertain to CMSE utilities.

Display Active Requests (AA, BA)

This command displays the first line of display requests issued by the PPs or the CPUs.

Command format:

```
AA

  or

BA
```

An example of the AA COMMAND display is shown in figure 3-53.

```
              CP0   PP 03   04   05   07   10   11
               *       D*   F    D*   D*   D*   I
                    PP 20   21   22   23   24   25   26   27   30   31
A=0000                 D*   D*   D*   D*   D*   D*   D*   D*   D*   D*

CP

PP

00  CE1
01  MEP
02  DSP
03  PCX RUNNING
04  HUNG
05  PCX RUNNNG
06  PCX RUNNING
07  PCX RUNNING
10  PCX RUNNING
11  IDLE
20  PCX RUNNING
21  PCX RUNNING
22  PCX RUNNING
23  PCX RUNNING
24  PCX RUNNING
25  PCX RUNNING
26  PCX RUNNING
27  PCX RUNNING
30  PCX RUNNING
31  PCX RUNNING
```

Figure 3-53. AA Command Display

Display Help Information (AI, BI)

This command displays a directory of CMSE command groups, commands within a group, or information about a specific command.

Command format:

```
AI,x

  or

BI,x
```

The x parameter indicates the type of information to be shown as follows:

x	Description
DIR or omitted	Display directory of the CMSE Information file. Figures 3-54 and 3-55 are examples of the directory as it appears below the A header.
command	Display information about the command specified. The command entry is any two or three letter mnemonic from one of the CMSE command groups shown in figures 3-54 and 3-55.

```
HELP INDEX 0000
  **   DIR    CMSE INFORMATION FILE    **

EXAMPLE- ENTER (AI,DMCH) FOR MAINTENANCE CHANNEL COMMANDS
DPP  - DIRECTORY OF PERIPHERAL PROCESSOR COMMANDS
DCM  - DIRECTORY OF CENTRAL MEMORY COMMANDS
DCS  - DIRECTORY OF CONTROL STORE COMMANDS
DSCM - DIRECTORY OF SOFT CONTROL MEMORY COMMANDS
DSPM - DIRECTORY OF SPECIAL MEMORY COMMANDS
DMCH - DIRECTORY OF MAINTENANCE CHANNEL COMMANDS
DCB  - DIRECTORY OF COMMAND BUFFER COMMANDS
DLIB - DIRECTORY OF DISK LIBRARY COMMANDS
DUTL - DIRECTORY OF UTILITY COMMANDS
DPCC - DIRECTORY OF PROGRAM CONTROL COMMANDS
KEY  - SPECIAL KEY USAGE
HELP - HOW TO USE INFORMATION FILE (I DISPLAY)
```

Figure 3-54. AI Command HELP Display

```
HELP INDEX 0000
  **   DIR    CMSE INFORMATION FILE   **

EXAMPLE- ENTER (AI,DMCH) FOR MAINTENANCE CHANNEL COMMANDS
DPP  - DIRECTORY OF PERIPHERAL PROCESSOR COMMANDS
DCM  - DIRECTORY OF CENTRAL MEMORY COMMANDS
DCS  - DIRECTORY OF CONTROL STORE COMMANDS
DSCM - DIRECTORY OF SOFT CONTROL MEMORY COMMANDS
DSPM - DIRECTORY OF SPECIAL MEMORY COMMANDS
DMCH - DIRECTORY OF MAINTENANCE CHANNEL COMMANDS
DCB  - DIRECTORY OF COMMAND BUFFER COMMANDS
DLIB - DIRECTORY OF DISK LIBRARY COMMANDS
DUTL - DIRECTORY OF UTILITY COMMANDS
DPCC - DIRECTORY OF PROGRAM CONTROL COMMANDS
CONS OR TERM OR PC - KEYBOARD USAGE
HELP - HOW TO USE INFORMATION FILE (I DISPLAY)
I4C  - NOTES ON I4C DIFFERENCES
```

Figure 3-55. AI Command HELP Display (AT478/AT481, AT511/AT512 only)

Display Register File (AS, BS)

This command causes 20 hexadecimal words, beginning at reg of the register file, to be displayed on either the left or right screen. The register file data is displayed in either hexadecimal or octal format, whichever is specified. If CS is not running, zeros will be displayed.

Command format for models 810, 815, 825, 830, 835, 840, 845, 850, 855, 860, and 960 only:

```
AS,Pnum,base,reg

    or

BS,Pnum,base,reg
```

where the following parameters apply:

num — CPU number selection. num is the CPU number. If omitted, the default CPU is assumed.

base — Indicates the data format as follows:

Omitted	Hexadecimal
H	Hexadecimal
O	Octal

reg — The first register to be displayed.

An example of the AS COMMAND or BS COMMAND display as it appears below the A or B header is shown in figure 3-56.

```
REGISTER FILE CPO
00  00000000 00000000
01  00000000 00000000
02  00000000 00000000
03  00000000 00000000
04  00000000 00000000
05  00000000 00000000
06  00000000 00000000
07  00000000 00000000
08  00000000 00000000
09  00000000 00000000
0A  00000000 00000000
0B  00000000 00000000
0C  00000000 00000000
0D  00000000 00000000
0E  00000000 00000000
0F  00000000 00000000
10  00000000 00000000
11  00000000 00000000
12  00000000 00000000
13  00000000 00000000
14  00000000 00000000
15  00000000 00000000
16  00000000 00000000
17  00000000 00000000
18  00000000 00000000
19  00000000 00000000
1A  00000000 00000000
1B  00000000 00000000
1C  00000000 00000000
1D  00000000 00000000
1E  00000000 00000000
1F  00000000 00000000
```

Figure 3-56. AS Command Display for Models 810, 815, 825, 830,835, 840, 845, 850, 855, 860, and CYBER 96X

Null Display (AN, BN)

This command clears the specified display below the header information and does not refresh that portion of the selected display until you select another display. This enables CMSE to execute faster because it is not constantly sending memory information to the display buffer.

Command format:

```
AN or BN
```

Do Not Monitor PP (DN)

This command deletes the selected PP from the CMSE system. CMSE does not monitor the downed PP nor accept any command other than UP for the PP. If you enter the command without a parameter, it is a no operation command.

For an AT511/AT512 IOU, channel 15 flag is cleared by the DN command. The UP command sets channel 15 flag as a signal that PP programs can again communicate with CMSE.

Command format:

```
DN

  or

DN,num or DN,/
```

where the following parameters apply:

num	One- or two-digit PP number of the selected PP. If omitted, the command is accepted, but no action is taken.
/	Substitute the last automatically assigned PP number.

Monitor PP (UP)

This command informs CMSE of the existence of the selected PP and clears the associated channel flag. It may be used to indicate that communication between the downed PP and CMSE is again possible. CMSE monitors and accepts commands for this PP if it is not idle.

For an AT511/AT512 IOU, channel 15 flag is used rather than the associated channel flag, and the flag is set by the UP command.

Command format:

```
UP,num

  or

UP,/
```

where the following parameters apply:

num	One- or two-digit PP number of the selected PP.
/	Substitute the last automatically assigned PP number.

Autoload Buffer Controller With Controlware From the Library Device (CW)

This entry causes controlware to be read from the library device and be autoloaded to the buffer controller on the channel specified by the command. The library device must not be on the same channel as the buffer controller to be autoloaded. The device being autoloaded cannot be placed on one of the channels corresponding to a PP or on one of the channels being used by CMSE. See Load from the CDA in this section for information on where microcode is being accessed for loading.

Command format:

```
CW,name,ch
```

where the following parameters apply:

name	Name of the controlware binary on library. Three to seven alphanumeric characters.
ch	Number of channel to which buffer controller to be autoloaded is connected. For an AT478/AT481 IOU, the CIO channel is not supported.

Clock Margins (CM)

This command sets the appropriate bit in the selected register to apply or remove the clock margins.

Command format for models 810, 815, 825, 830, 835, 840, 850, 855, 860, and 990 only:

```
CM,Pnum,el,mgn
```

where the following parameters apply:

Pnum — CPU num selected. If omitted, the default CPU is assumed.

el — Mainframe element selection.

0 or I	IOU
1 or M	Memory
2	CPU0

mgn — Determines the type of clock margins as follows:

N	Narrow clock pulse width
W	Wide clock pulse width
blank	Normal margins

Command format for model 960 only:

```
CM,el,mgn
```

where the following parameters apply:

el	Mainframe element selection.
	0 or I — IOU
	1 or M — Memory and CPUs
mgn	With 0 or I selected determines these types of clock margins:
	N — Narrow clock pulse width
	W — Wide clock pulse width
	blank — Normal margins
	With 1 or M selected determines these types of clock margins:
	N — Narrow clock pulse width, normal rate
	W — Wide clock pulse width, normal rate
	F — Normal margins, fast rate
	S — Normal margins, slow rate
	NF — Narrow clock pulse width, fast rate
	NS — Narrow clock pulse width, slow rate
	WF — Wide clock pulse width, fast rate
	WS — Wide clock pulse width, slow rate
	blank — Normal margins, normal rate

Log Errors (LE)

This command causes critical errors detected in the selected element to be displayed and written to the dayfile. The el parameter selects which mainframe element errors are to be logged when error logging is enabled. If el is omitted, errors are logged for all elements.

Command format:

```
LE,el,cond
```

where the following parameters apply:

el	Mainframe element selection. If omitted, errors are logged for all elements.

0 or I	IOU
1 or M	Memory
2	CPU0
Pnum	CPU num selected. If omitted, the default CPU is assumed.

cond	Status of error logging.

ON	Enable error logging.
OFF	Disable error logging.

Activate Channel (*AC)

This command activates a channel.

Command format:

```
*ACn
```

where the following parameter applies:

n	Channel number. On an AT478/AT481 IOU, the CIO channel is not supported.

Deactivate Channel (*DC)

This command deactivates a channel.

Command format:

```
*DCn
```

where the following parameter applies:

n	Channel number. On an AT478/AT481 IOU, the CIO channel is not supported.

Send Function to Channel (*FC)

This command sends a specified function on a channel.

Command format:

```
*FCn,f
```

where the following parameter applies:

n	Channel number. On an AT478/AT481 IOU, the CIO channel is not supported.
f	Function to be sent on channel (one through four octal digits). Default value is 0.

NOTE

A function of 1700 is a 6681 master clear function.

Input to Pseudo-A Register (*IA)

This command inputs a data word to pseudo-A register from channel n. The system displays pseudo-A register on the third line of the right-screen display header.

Command format:

```
*IAn
```

where the following parameter applies:

n	Channel number. On an AT478/AT481 IOU, the CIO channel is not supported.

Output Data Word on Channel (*0A)

This command outputs a data word on channel n.

Command format:

```
*OAn, x
```

where the following parameters apply:

n	Channel number. On an AT478/AT481 IOU, the CIO channel is not supported.
x	Data word output (one to four characters, right-justified). For AT511/AT512 IOU only, one to six characters, right-justified.

Load Monitor Overlays to CM (*OV)

This command reads the monitor overlays from the disk or tape library and writes them into CM beginning at address adrs. The monitor then uses the overlays from CM to speed up CMSE execution. This command is not executed if CM is already in use by CMSE. If the overlays cannot be read from CM, a deadstart is required.

For an AT511/AT512 IOU only, CMSE automatically reverts to disk to obtain overlays if overlays loaded to CM by the *OV command are wiped out. CMSE sends the message `CM OVERLAYS CORRUPTED, REVERTING TO DISK`. If a dedicated load device is present, the *OV command is accepted regardless of CM usage option selection. If the CM usage option is enabled and a *OV command entered with the adrs field blank, overlays are reloaded to their original position in the protected upper megabyte of CM.

Command format:

```
*OV,base,adrs
```

where the following parameters apply:

base	Omitted	Hexadecimal
	H	Hexadecimal
	O	Octal
adrs	First word address in CM for the monitor overlays.	

NOTE

Only the first word of an overlay is checked for validity when it is read from CM. Therefore, caution must be taken when using this feature so as not to overwrite part of a monitor overlay.

Load Primary/Secondary Console Driver (LR)

This command loads a display driver for primary or secondary consoles from the disk or tape library and writes it to PP memory. CMSE installs or replaces the current display driver with either the CC545 display driver (DSP), the CC634B, 752-compatible, or the CC598-A/B (without the AEP) with DSQ4. An additional dual screen driver (DSD) also supports the CC598-A/B if it has the AEP, terminal type 2. The dual screen driver (DSD) supports both primary and secondary consoles (CC598) if both have AEP.

NOTE

If a CC598-A/B is the primary console, and if the remote console (which is acting as the controlling console) wants to invoke both primary and secondary consoles, the primary console must display the CONSOLE MAIN MENU before the LR command can be executed.

If the deadstart is from a CC598-A/B, the CMSE display is a single screen display (DSQ4). If the CC598-A/B has AEP, the user must invoke the command LR,1520 in order to utilize the dual screen driver.

Command format:

```
LR,ntp

  or

LR,ntp,ntp
```

where the following parameters apply:

n	Channel number (10_8 or 15_8).
t	Terminal type.
	0 CC634B, CC545, or CC598-A/B without AEP
	1 752-compatible
	2 CC598-A/B with AEP
p	Port number.
	0 CC545
	0 or 1 CC634B, 752, or CC598-A/B

The TPM is assumed to be equipment 7 on channel 15_8. The CC545 is assumed to be on channel 10_8. If the current primary console is not a CC545 and if a CC545 is specified anywhere in the LR command, it automatically becomes the primary console. Otherwise, the first ntp parameter in the LR command defines the primary console.

Lock CMSE Area of CM From Keyboard Commands (LK)

This command causes CMSE to lock out the area of CM reserved for CMSE use from any modification with a keyboard command. Any keyboard command attempting to enter data into the protected area of CM is aborted and the system displays an error message on the keyboard error message line. If CM is not in use by CMSE, the system displays an informative message.

Command format:

```
LK
```

Unlock CMSE Area of CM for Keyboard Commands (UL)

This command causes CMSE to allow processing of keyboard commands which enter data into the CMSE area of CM when CM is in use. If CM is not in use, the system displays an informative message.

Command format:

```
UL
```

Revert to Monitor Overlays From Disk (*RV)

This command drops use of CM monitor overlays and reverts to loading overlays from disk. If a dedicated load device is not present and the CM usage option is enabled, this command is rejected with message CANNOT CHANGE CM USAGE WITHOUT DEADSTART.

Command format (AT511/AT512 IOU only):

```
*RV
```

Program Control Commands

Program control commands alter parameter bits contained in either of two parameter words. The address of the first parameter word is found in PPM address 0005. The address of the second parameter word is the first parameter word address plus 1. If bit 2^{15} is set in the first parameter word, CMSE interprets the commands associated with the first parameter word and sets or clears the appropriate parameter bit. If bit 2^{15} is not set, CMSE does not interpret the command, but leaves it in the keyboard buffer as a job command to be passed to any PP requesting keyboard/display data. If bit 2^{15} is set in both the first and second parameter words, CMSE interprets the commands associated with the first and second parameter words and sets or clears the appropriate parameter bit.

These commands allow programs to use a common command set. Only the codes that have their corresponding bit set are displayed. They are displayed whenever the using program makes a 43 call to display information and bit 2^8 is set in word 5 of the call block. The codes are updated when a program control command is accepted or the B screen changes from a memory display to a job display. However, it is not mandatory that a program use any of this capability not applicable to the program. Refer to test descriptions for application. The commands, when entered, clear and set the bits regardless of their use by the program. Command formats, descriptions, and parameter bits associated with each command are shown in the table below. In the table, (5) is interpreted as the contents of PPM address 0005.

NOTE

When using the CSM/SSM commands in conjunction with the channel reservation flag, the user programs must always clear bit 2^{10} upon detection of the channel flag set condition or CMSE may not be able to regain communication with the PP.

Format	Definition	Bit	Word
CST or SST	Clear/Set Stop at end of test	2**0	(5)
CSS or SSS	Clear/Set Stop at end of section	2**1	(5)
CSB or SSB	Clear/Set Stop at end of subsection	2**2	(5)
CSC or SSC	Clear/Set Stop at end of condition	2**3	(5)
CSE or SSE	Clear/Set Stop on error	2**4	(5)
CLE or SLE	Clear/Set Log errors	2**5	(5)
CRT or SRT	Clear/Set Repeat test	2**6	(5)
CRS or SRS	Clear/Set Repeat section	2**7	(5)
CRB or SRB	Clear/Set Repeat subsection	2**8	(5)
CRC or SRC	Clear/Set Repeat condition	2**9	(5)
CSM or SSM	Clear/Set Scope mode	2**10	(5)

Format	Definition	Bit	Word
CQL or SQL	Clear/Set Quick look	2**11	(5)
CDR or SDR	Clear/Set Bypass all messages (overrides bit 2**13)	2**12	(5)
CDE or SDE	Clear/Set Display error message	2**13	(5)
	(RESERVED)	2**14	(5)
	Must be set via program in order to set or clear parameter bits associated with PP word (5) or (5)+1.	2**15	(5)
	(UNUSED)	2**0 2**6	(5)+1
CSR or SSR	Clear/Set Scope mode repeat	2**7	(5)+1
CCE or SCE	Clear/Set Report conditions errors	2**8	(5)+1
	(UNUSED)	2**9 2**11	(5)+1
CSI or SSI	Clear/Set Stop at end of iteration	2**12	(5)+1
CRI or SRI	Clear/Set Repeat iteration	2**13	(5)+1
CAB or SAB	Clear/Set Abort subsection on errors	2**14	(5)+1
	Must be set via program in addition to bit 2**15 of PP word (5), in order to set or clear parameter bits associated with PP word (5) or (5)+1.	2**15	(5)+1

CMSE Standalone Test Loader

Any 16-bit IOU diagnostics may be run in stand-alone mode by entering the diagnostic name via the keyboard at the INITIAL CMSE display. Refer to Tape Deadstart Display in section 2 for the INITIAL CMSE display.

Command Buffer Program Library Construction and Maintenance 4

Command Buffer Program Library Construction and Maintenance 4

This section describes the procedures for constructing and manipulating command buffers and programs on the disk library. The procedures for creating a command buffer, modifying a command buffer, and adding programs to the disk library are described.

Creating a Command Buffer

A command buffer is a series of keyboard commands residing on the disk library. The system monitor executes the command buffer as a group. Each command buffer has an assigned name to identify it in the library.

Command buffers can be added to the library by doing one of the following:

- Typing entries at the keyboard
- Using TDX

NOTE

If a disk installation is new and no command buffers exist, enter *DB * to clear the command buffer area.

Creating a Command Buffer at the Keyboard

Command buffers can be created at the keyboard by performing the following steps.

1. Specify a name for the command buffer on the edit file by entering an AE,name or BE,name command.

2. Enter IN,0,command as the first buffer entry.

3. Press the **Forward** key (CC545 only), enter the next command, and repeat this step until all buffer commands are entered.

4. Press the **Erase** key (CC545 only) after you enter the last command, or enter the last command using **(CR)**.

5. Enter a GO * command to verify a command buffer entered from the keyboard by executing it before it is placed on the library.

   If you specify a command incorrectly, use the DE command to delete or modify the command and repeat step 5.

6. Enter a *WB command to copy the completed command buffer from the edit display to the library.

Refer to Command Buffer Commands in section 3 for a complete listing and explanation of command buffer commands.

Adding Comments to a Command Buffer

Comments may be made on the same line as a command buffer command. The entire line must not exceed 60 characters including the Common Maintenance Software Executive (CMSE) command. If comments are to be added, all parameters for the CMSE command must be accounted for either by a value or by successive commas. For example:

```
*WP,,5,DMP,,,COMMENT-WRITE DMP TO DISK
```

Command Buffer Example

The following example shows how a command buffer may be used to change default parameters and start execution of a test.

```
$$ FT8
DP5                  DEADSTART PP5.
KP5,,,               CLEAR PP5 MEMORY.
CP5,FT8,,,,          LOAD FT8 INTO PP5.
EP5,1001,1000        SET REPEAT TEST BIT.
EP5,1002,0100        CHANNEL 1, EQUIPMENT 0.
PP5,,                ASSIGN DISPLAY TO PP5.
RU5,101              RUN FT8 AT ADDRESS 101.
TB                   TERMINATE BUFFER.
```

Command Buffer Modification

Commands may be added or deleted from a command buffer by performing the following steps.

1. Enter an AE or BE command and the name assigned to the command buffer to be modified.

2. Enter an IN command to insert a new command and/or enter a DE command to either delete or change a specific command from the command buffer.

3. Enter a *DB command and the name assigned to delete the previous command buffer.

4. Enter a *WB command to copy the modified command buffer from the edit file to the library.

Adding Programs to the Disk Library

Programs can be added to the disk library by performing the following procedure or by using the TDX utility program described in section 6 of this manual.

Enter a *WP, *WC, or *WK command at the display station keyboard to copy existing programs from a peripheral processor (PP) memory, central memory (CM), or control store (CS), respectively.

Adding Programs From a PP, CM, or CS

To copy an existing program from PP memory to the monitor or program disk, enter the *WPn command at the keyboard.

To copy a program from CM to the disk, enter the *WC command.

To copy a program from CS to the disk, enter the *WK command.

List Command Buffers Provided on MSL Tape

The following procedure can be used to list the contents of all of the command buffers that exist on a released MSL tape. The procedure is written for use with an operating system.

NOTE

The contents of the command buffers can also be found on microfiche.

1. Catalog the MSL tape using the following job deck.

```
Job control cards
Accounting information
REQUEST,MSL,NT,D=PE,F=SI,LB=KU,PO=R,VSN=MSL01  } for NOS or
REQUEST,MSL,NORING,VSN=MSL01                   } for NOS/BE
CATALOG,MSL.
7/8/9
```

2. Examine the resulting output and determine the record number of the first command buffer. The record numbers are printed on the left side of the output listing, and a command buffer can be recognized by a $$ (double dollar sign) as the first two characters of the name. Subtract one from the record number of the first command buffer and use the resulting number as the parameter on the SKIPR card of the next sequence.

3. Extract the command buffer listing using the following job deck.

```
Job control cards
Accounting information
REQUEST,MSL,NT,D=PE,F=SI,LB=KU,PO=R,VSN=MSL01  } for NOS or
REQUEST,MSL,NORING,VSN=MSL01                   } for NOS/BE
SKIPR,MSL,xxx[1]
COPYSBF,MSL,OUTPUT[2]
7/8/9
```

4. The resulting printout lists all of the command buffers on the MSL tape.

1. Skips the number of records specified by xxx (where xxx is the record number parameter from step 2) which positions the MSL tape at the beginning of the first command buffer.

2. Copies the command buffers from the tape to the output file and shifts the data one column to the right to avoid sending the first character of each line as a paper motion control character when the output file is printed.

Diagnostic Interface Routines 5

Diagnostic Interface Routines 5

There are three programs in MSL 15X that act as interfaces to other tests and utility programs within the MSL, but are not completely transparent to the user: the virtual level executive (VEXC), the peripheral processor (PP)-based central processing unit (CPU) monitor program (EXC) and the DEMOT executive. A fourth program, the diagnostic executive (DEX), is common to most PP-based maintenance software. The use of DEX is completely transparent to the user.

The cache initialization binary (OCCI) is used to initialize the model 990 cache tag memory prior to running diagnostics.

Virtual Level Executive (VEXC)

VEXC is a CPU-based program enabling executive state CPU tests to execute in a virtual addressing mode. It is the primary communication link between the CPU tests and Common Maintenance Software Executive (CMSE). In addition, VEXC processes traps and interrupts encountered during test execution. It keeps track of instruction failures resulting from a process interval timer (PIT) or system interval timer (SIT).

Program Description

VEXC provides the virtual environment in which the tests execute. Following are the six elements that make up VEXC.

Page Table

The page table enables the conversion from process virtual addresses (PVAs) to real memory addresses (RMAs). The page table is loaded at RMA 0.

Segment Table

A segment table exists for the monitor process state and the job process state. It is used in common by VEXC and the loaded CPU test. The segment table is used to reference the page table.

Monitor Process State Exchange Package (MPS)

The MPS establishes the environment for execution in monitor state. If testing is done in monitor state, it is necessary to execute an additional CMSE command to copy the Job Process State Exchange Package (JPS) to the MPS area prior to test execution.

Job Process State Exchange Package (JPS)

The JPS establishes the environment for test execution. An exchange from monitor process state starts test execution.

Monitor Exchange Routine

When the CPU is initialized, the monitor exchange routine does an exchange to job process state to start test execution. On any subsequent return to monitor process state, this routine examines the monitor condition register (MCR) monitor mask (MM), user condition register (UCR), and user mask (UM) to determine the cause of the exchange and returns to job process state or outputs an error message.

Trap Interrupt Handler

If a Job mode trap condition exists, control is passed to the trap handler. The trap handler examines the UCR and the UM to determine the cause of the exchange. For an expected trap, control is returned to the executing test. For an unexpected trap, test execution stops and the system displays an error message in the B display (right screen).

Loading Procedure

VEXC is assembled with a test using the virtual machine assembler. CMSE loads the linked binary (the combination of VEXC and the CPU test) beginning at RMA zero. Any changes required to satisfy the initialization requirements of the individual tests are made at load time with the CMSE EC command (typically in a command buffer).

Parameters

During initialization of VEXC the user is requested to set up the following monitor options.

Option	Description
Page faults	If enabled, VEXC invalidates all the job pages before the initial exchange to job state.
Map purges	If selected and a page table search without find causes an exchange to monitor state, VEXC executes a purge instruction purging map before returning control to job state.
Cache purges	If enabled and a page table search without find causes an exchange to monitor state, VEXC executes a purge instruction purging cache before returning control to job state.
Limited working set size	If selected, repetitive page faulting occurs. The user is asked to enter the maximum number of pages to be valid at any given time. When VEXC recognizes the working set size has been met, all job pages are invalidated again.
32 entry page table search	If selected, VEXC sets up the page table so that there is one valid entry for every 32 entries with the continue bit set between entries. The small SIT option must be used with this option.

Informative Messages

VEXC presents the following message if Not Monitor Default and Not Monitor Bypass are selected in the CPU test. Other informative messages are displayed if the operator answers the question in the last line of the message.

```
          V E X C  MONITOR PROCESSOR  X    VERSION X.X  MM/DD/YY

VEXC AUTOMATICALLY KEEPS A HISTORY OF ALL EXCHANGE INTERRUPTS
AND MONITOR AND JOB TRAP INTERRUPTS.  IT ALSO KEEPS A RECORD OF
RESPONSE TIMES WHICH EXCEED 25 MICRO-SECONDS AND HANG CASES
WHERE AN INSTRUCTION FAILS TO COMPLETE AS A RESULT OF A PIT/SIT

TO DISPLAY HISTORY BUFFERS ENTER  -  AH,XXXX

THE BUFFERS ARE NAMED AS FOLLOWS

  EXCH - EXCHANGE INTERRUPTS        RESPONSE - RESPONSE TIME
  MON TRAP - MONITOR TRAPS          P1  ERRS - DUE/SOFT ERRORS
  JOB TRAP - JOB TRAPS              SIT HAND - SIT HAND CONDS.
  UM  PTBL - USED/MODIFIED CHECK    PIT HAND - PIT HANG CONDS.

DO YOU WANT TO SEE A DESCRIPTION OF THE BUFFERS (Y/N)
```

Monitor processor x	The CPU identification of the processor running at this time.
Version X.X	The version number of VEXC is also displayed in the test running message.
AH,XXXX	The word address where these buffers start.

Error Messages

The following are standard error messages displayed due to a parameter stop.

```
PAGE FAULT  - PAGE TABLE FULL -
HALT CONDITION
   MCR/UCR = xxxxyyyy  UTP = xxxxxxxxxxxxxxxx
                   P (PVA) = xxxxxxxxxxxxxxxx
                     (RMA) = xxxxxxxx
SCOPE LOOP (Y/N) -
```

The previous message indicates that the page table is full. The CPU is halted in the monitor state. The user is asked if a scope loop should be initiated.

```
PAGE FAULT  - PAGE ALREADY VALID
   MCR/UCR = xxxxyyyy  UTP = xxxxxxxxxxxxxxxx
                   P (PVA) = xxxxxxxxxxxxxxxx
                     (RMA) = xxxxxxxx
SCOPE LOOP (Y/N) -
```

The previous message indicates that the CPU has taken a page fault on a valid page. The user is asked if a scope loop should be initiated.

```
PAGE FAULT  - PAGE NON-EXISTENT
   MCR/UCR = xxxxyyyy  UTP = xxxxxxxxxxxxxxxx
                   P (PVA) = xxxxxxxxxxxxxxxx
                     (RMA) = xxxxxxxx
SCOPE LOOP (Y/N) -
```

The previous message indicates that the CPU has taken a page fault on a page that does not exist. The user is asked if a scope loop should be initiated.

```
UNEXPECTED EXCHANGE
   MCR/UCR = xxxxyyyy  UTP = xxxxxxxxxxxxxxxx
                   P (PVA) = xxxxxxxxxxxxxxxx
                     (RMA) = xxxxxxxx
SCOPE LOOP (Y/N) -
```

The previous message indicates an unexpected exchange from job state to monitor state has occurred. The user is asked if a scope loop should be initiated.

```
UNEXPECTED TRAP IN MONITOR MODE
   MCR/UCR = xxxxyyyy  UTP = xxxxxxxxxxxxxxxx
                   P (PVA) = xxxxxxxxxxxxxxxx
                     (RMA) = xxxxxxxx
SCOPE LOOP (Y/N) -
```

The previous message indicates that an unexpected trap has occurred in monitor mode. If the user observes that the untranslatable pointer (UTP) on the display is not the same as the UTP in the processor, then the processor at the time of the trap was using the monitor segment table, segment 8, to access a page in the job's address space. The job PVA causing the trap is the UTP displayed on the screen. The user is asked if a scope loop should be initiated.

```
UNEXPECTED TRAP IN JOB MODE
   MCR/UCR = xxxxyyyy  UTP = xxxxxxxxxxxxxxxx
                   P (PVA) = xxxxxxxxxxxxxxxx
                     (RMA) = xxxxxxxx
SCOPE LOOP (Y/N) -
```

The previous message indicates that an unexpected trap has occurred in job mode. The user is asked if a scope loop should be initiated.

```
SIT POSITIVE,  VALUE = xxxxxxxx
```

The previous message warns the user that an exchange to monitor was caused by a SIT fire and the SIT was found to be positive and nonzero.

```
SOFT ERROR P = xxxxxxxxxxxxxxxx
  MCR = xxxx
 MCEL = xxxx
```

VEXC logs the previous message in the model 810/830 CMSE180 dayfile when an exchange to monitor state was caused by bit 48 or 62 being set in MCR.

```
SOFT ERROR P = xxxxxxxxxxxxxxxx
  MCR = xxxx
 CCEL = xxxxxxxx          MCEL = xxxxxxxxxxxxxxxx
```

VEXC logs the previous message in the model 835 CMSE180 dayfile when an exchange to monitor state was caused by bit 48 or 62 being set in MCR.

```
SOFT ERROR P = xxxxxxxxxxxxxxxx
  MCR = xxxx
 PFS0 = xxxxxxxxxxxxxxxx
 PFS1 = xxxx
 PFS2 = xxxxxxxxxxxxxxxx
 PFS3 = xxxx
 PFS4 = xxxxxxxxxxxxxxxx
 PFS5 = xxxx
 PFS6 = xxxxxxxxxxxxxxxx
 PFS7 = xxxx
 PFS8 = xxxxxxxxxxxxxxxx
 PFS9 = xxxx
```

VEXC logs the previous message in the model 840/860 CMSE180 dayfile when an exchange to monitor state was caused by bit 48 or 62 being set in MCR.

PP-Based CPU Monitor Program (EXC)

The CPU monitor (EXC) program loads and controls all CYBER 170 state CPU tests. EXC loads and controls from one to four CPU tests for each CPU from a PP.

Hardware Requirements

EXC requires one CPU, central memory (CM), and a monitor library load device.

Software Requirements

EXC loads and runs under control of the system monitor.

Loading Procedure Using a Command Buffer

The following command buffer loads EXC, then loads and runs any EXC controlled CPU test.

Command	Description
LT,EXC	Load EXC into first available PP.
SQ75,1,/	Wait until EXC initializes.
L,xxx,yyy,zzz,aaa	Load CPU test. One to four tests may be specified.
SQ75,1,/	Wait until tests are loaded.
SP	Start tests executing.

Loading Procedure Calling EXC From the Library

The following format is used when calling EXC from the library:

```
LT,EXC,p0,p1,p2,p3
```

where the following parameters apply:

p0=0	CPU 0.
p0=1	CPU 1.
p1=0	EXC loads tests in CM, starting at location 10000_8.
p1=1	Allows the two CPUs to share CM.
p2	RA/100_8.
p3	FL/100_8.

Display Commands

Display commands for EXC are described as follows.

Help Display (H)

The H command describes the commands and parameters for EXC.

Normal Running Display (N)

The N command provides the running test status display.

The displayed information is as follows:

```
        MODE = RUN              AVAILABLE FL = xxxxxx
        BKP = xxxxxx            EXCHANGE RATE = xxxx

  0     mne P=xxxxxx            EX = xxx
            RA=xxxxxx           FL = xxxxxx

        (message buffer)        Three lines, 46 octal characters per line

  1     mne P=xxxxxx            EX = xxx
            RA=xxxxxx           FL = xxxxxx

        (message buffer)        Three lines, 46 octal characters per line

  2     mne P=xxxxxx            EX = xxx
            RA=xxxxxx           FL = xxxxxx

        (message buffer)        Three lines, 46 octal characters per line

  3     mne P=xxxxxx            EX = xxx
            RA=xxxxxx           FL = xxxxxx

        (message buffer)        Three lines, 46 octal characters per line

        (error message buffer)
```

Fast Running Display (T)

The T command allows the user to shorten the display time of the running test status. A second T command brings back the original display.

EXC Keyboard Commands

The following are the keyboard commands for EXC.

Load CPU Tests (L)

The following is the load CPU tests command:

```
L,mne,mne,mne,mne
L,mne,mne,mne,mne,ra
L,mne
L
```

where the following parameters apply:

mne	Name of CPU test (three or four alphanumeric characters). If the mne parameters are absent, four CPU tests are loaded.
ra	The load address for the start of the test (one to six octal digits). The default address is 10000_8.

If from two to four tests are loaded, EXC operates in a in a multiprocess mode.

Set Field Length for EXC (AFL) or Test

The following command sets total available field length for EXC or test:

`AFL,v`	For EXC
`CFL,x,v`	For test

where v is the field length (one to six octal digits) and x is the control point number of the test.

Tests are loaded beginning at the absolute CM location specified by v.

Set Exchange Rate (EXR)

The following command sets the exchange rate for EXC:

```
EXR,u
```

where u is one to four octal digits; u is preset to 20_8 (an exchange every 40 microseconds).

Restart CPU Test (R)

The following command restarts the CPU test(s):

```
R,x
```

where x is the control point number of the test. If no x parameter is entered, the current exchange package of each test is reset.

Start Selected Test (G)

The following command restarts the test previously stopped by an S,x command:

```
G,x
```

where x is the control point number of the test.

Start CP (Space Bar)

Pressing the **Space Bar** causes EXC to exchange the CP with one of the following packages.

- If you select Run or Test mode, exchange is done with the input package.
- If you select Step mode, exchange is done with the last output package.

Stop CP (S)

The S command causes EXC to exchange the CP with an idle package and halt execution of the test(s) currently selected.

H Display Toggle Commands

The following commands toggle on/off flags in the H display.

Set Auto Exchange Rate Flag (A)

The A command automatically sets the proper exchange rate if the flag is on.

Set DDP/ECM Flag (B)

The B command allows DDP to run simultaneously with ECM when the flag is turned on.

Set Error Stop (E)

The E command causes EXC to stop on a CPU error if turned on, and not to stop on a CPU error if turned off. The default setting is on.

CPU Manipulation Commands

The following commands manipulate the CPU.

Set Exchange Address (EXK)

The following command sets the exchange address to the value v:

```
EXK,v
```

where v is the exchange address (one to six octal digits). Default is 400_8.

Set Breakpoint Address (BKP)

The following command sets the breakpoint address to the value v:

```
BKP,v
```

where v is the breakpoint address (one to six octal digits). The instructions at address v are saved and replaced with a program stop instruction.

Set Run Mode (RUN)

The following command causes the CPU to run until the breakpoint is reached.

```
RUN
```

When the breakpoint is reached, the CPU is exchanged out and the breakpoint address is restored.

Set Test Mode (TEST)

The following command permits looping on an instruction or a series of instructions.

```
TEST
```

The P register is continually monitored for breakpoint address. When the breakpoint address is reached, the CPU is exchanged out. The same program is reexecuted by exchanging it with the original input exchange package.

Set Step Mode (STEP)

The following command permits an operator to monitor the execution of consecutive CPU instructions.

```
STEP
```

EXC sets up the Step mode in which the P register is continually monitored for the breakpoint address. When the breakpoint address is reached, the CPU is exchanged out. When the CPU is restarted (space bar), the same program is reexecuted by exchanging it with the original output exchange package, and the breakpoint address is incremented by 1 each time the **Space Bar** is pressed.

Clear Breakpoint for CPMTR (CBP)

The following command clears the CP monitor (CPMTR) capability to loop on error central exchange jump (CEJ) conditions.

```
CBP
```

Set Breakpoint for CPMTR (SBP)

The following command enables CPMTR to loop on an error CEJ condition.

```
SBP
```

A user program, with monitor flag clear, and exiting with any mode error, is restarted using its input exchange package.

Cache Initialization Binary (OCCI)

OCCI is loaded into the model 990 cache tag memory from the CIP command buffers prior to running diagnostics. It initializes the memory without having to deadstart the system.

Loading Procedure

OCCI is loaded with command WS, OCCI, OCC.

DEMOT Executive (DEMOT)

DEMOT runs under user control with the CMSE. The support function within the CMSE system includes a PP executive that provides a user interface. DEMOT provides a compiler that converts the source language diagnostics into executable PP code and PP driver/product overlays that execute the compiled code.

DEMOT is composed of the following seven elements.

1. A PP executive that controls the compiler and the PP driver and provides a user interface.

2. A compiler that compiles non-I/O language commands, low-level language commands, and high-level language commands to generate executable binary PP code. Non-I/O commands are used with both levels of language to control the program and manipulate data buffers and operating registers.

3. A PP driver that executes non-I/O statements.

4. Product overlays to the PP driver that execute the low-level and high-level language commands.

5. Non-I/O language commands that control module program flow and data in the PP and which apply to both language levels. These commands are detailed in the MALET Reference Manual.

6. Low-level language commands for channel interfaces and controllers that are not device oriented. These include commands such as CONNECT, STATUS, OUTPUT, and FUNCTION. This level of language is used to write diagnostics designed to detect and isolate errors and allows the repair of controller and channel interfaces. These commands are detailed in the MALET Reference Manual.

7. High-level language commands for specific hardware families. These commands include non-I/O statements and such device-oriented I/O statements as REWIND, ENDFILE, and so forth for tapes; and PRINT, EJECT, and so forth for printers. This level of language is used to write diagnostics designed to detect and diagnose errors and allows the repair of peripheral units. These commands are detailed in the MALET Reference Manual.

Executive

The executive is loaded to a PP in the CMSE system and is then the master of the other components. Input data to the executive is in the form of directives, which are executed by the executive, or source code, which is saved for the compiler.

The executive:

- Acts upon executive directives.
- Saves source code lines in CM for the compiler.
- Calls and controls the CPU compiler.
- Drives the DEMOT displays.
- Calls and controls the driver and its product overlays.
- Honors PP requests for assistance during the execution of modules.

The DEMOT executive provides the following capabilities for input of directives and source code.

- Directives may be input from the keyboard, card reader, or a CMSE command buffer.
- Source code may be input from the keyboard, card reader, or, along with a precompiled module, loaded from the Maintenance Software Library (MSL).
- Precompiled modules can be loaded from the MSL program library and executed.
- A card deck may be read on the card reader to input executive directives and source code. DEMOT output is written to a disk library file named OUTPUT.

Data for the DEMOT display is organized in the form of pages that can be displayed upon user request. Page displays are 60 decimal characters per line wide and 40 decimal lines deep.

The PP messages display is a page derived from data generated by modules running in the PP. The display buffer may contain up to 40 decimal messages. Each line is directly addressed by the module.

In addition to the message capability, a graphics capability allows the programmer to generate a plot within the same display area, intermixed with messages. Using the PLOT command, an alpha character asterisk (*) is added to the existing display area on a specified line and character position.

The BLANK command allows the programmer to blank fill the PP display contents. The PICTURE command sends the current display contents to the output file.

Input data to the executive that are not recognized as directives are assumed to be source code and are saved for later compilation. Figure 5-1 shows data flow into the MSL library.

Typically, a number of modules are required to form a total diagnostic. The user has the option to run specific modules in a specific sequence, plus options to loop on a single module, or to run a series of modules.

Source code can be entered, compiled, and saved for execution.

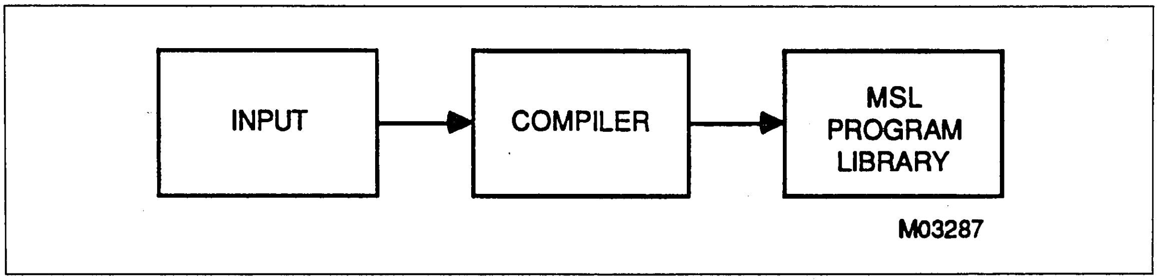

Figure 5-1. Data Flow Into the MSL Program Library

All modules executed by the PP driver are obtained from the MSL program library by the PP executive (figure 5-2).

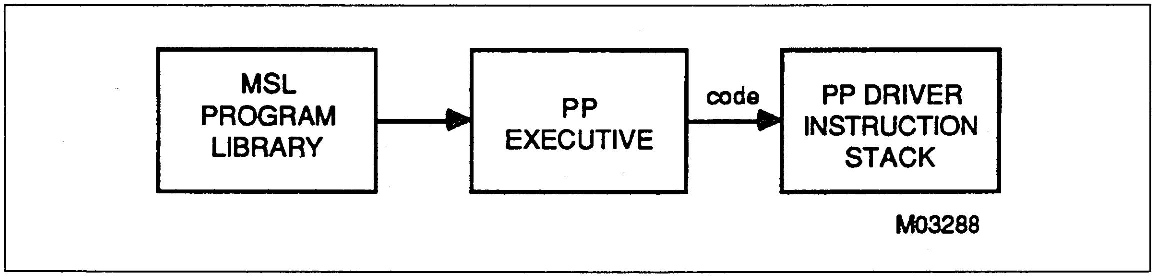

Figure 5-2. Data Flow Out of the MSL Program Library

The complete module is read and the source code, if present, may be displayed and/or modified and recompiled. If this module is run, the executable code is executed in the PP driver. If there is any compiled data, it is passed to the PP for access by the module.

A variety of capabilities exists for obtaining/generating data within a module to test I/O device data paths.

- Data defined in the DATA statement can be carried with the module binary.
- Executable statements (DUP, RANDOM, and the replacement statements) can be used to dynamically generate data.
- Data can be obtained from a user-supplied scratch file with the SYSRD command.

The output-of-results mechanism within DEMOT allows the module to accumulate messages in the PP display and control the time at which a picture of this display is sent to the output file.

Compiler

The compiler compiles the source code and generates an executable binary file. It can also produce a listing of the source code and binary PP code which is sent to the output file. The compiler may be used to compile only if running CMSE disk based.

PP Driver

The maintenance language driver (MLD), also known as the PP driver, executes PP code in the form of modules that have been compiled by the DEMOT compiler. These modules are diagnostic subroutines used to verify the operation of a peripheral device or to detect and aid in error isolation and repair of the device.

The driver is called into execution with a CMSE RU command and remains in execution until a DROP command is processed by the executive. Module execution begins when a RUN command is processed by the executive and continues until one of the following conditions occurs.

- A STOP directive is processed by the executive and passed to the driver. The PP stops when a logical breakpoint is reached (that is, a RES or REL sequence is executed).
- An END statement is executed by the PP driver. This causes the PP and the executive to enter their respective idle routines.
- The PP driver detects that more than 4 seconds have elapsed since the last attempted release of the assigned I/O channel to the CMSE system.

PP Product Overlay

The PP product overlay is a specialized overlay to the basic PP driver. Its function is to drive specific I/O devices.

Non-I/O Commands

The non-I/O DEMOT language commands apply to both language levels. The primary purpose of these commands is to provide means to control the flow of the program within a module and the control of data within the operating registers and buffers in the PP. The syntax details are in the MALET Reference Manual.

Low-Level I/O Commands

Low-level I/O commands apply to all 3000 and 6000 channels, as well as to buffer controller devices on 6000/CYBER 70/CYBER 170 channels. This level of language is used primarily to test I/O channel interfaces and controllers and the details may be found in the MALET Reference Manual.

High-Level I/O Commands

High-level I/O commands are designed for each specific hardware family. This level of language is used primarily to test peripheral units. High-level I/O commands are detailed in the MALET Reference Manual.

Structure and Organization

In addition to the elements of DEMOT, it is necessary to understand the structure and organization. The structure and organization consist of the module, registers and buffers, and data organization.

Module

The smallest block of code DEMOT can execute is a module. A single module can be executed, or a series of modules can be linked and executed in a defined order.

A module is a maximum of 120 decimal lines of source code consisting of an identification division, a data division, an executable division, and a termination division. A slash (/) statement, documenting the purpose/use of the module, is allowed anywhere between identification and termination divisions. The four divisions are:

1. The identification division names the module and describes the device type it is written for and the specific device codes supported.
2. The data division is optional and may contain:
   a. DATA statements, defining data to be used within the module.
   b. FORMAT statements, defining normal/error messages for use within the module. A maximum of eight microsubstitutions of values may be specified within each FORMAT statement, and a maximum of 24 FORMAT statements is allowed per module.
   c. EQUATE statements, equating a user-specified name to an octal or decimal value. A maximum of 20 names is allowed. The equated value may be used within the module for any variable for which an octal or decimal value is legal.
   d. BASE statements, identifying whether numbers defined within source code statements are octal or decimal.
3. The executable code division contains the commands that create executable PP code for the PP driver. Language levels cannot be mixed within the same module.
4. The termination division ends the module and specifies the entry point for the module.

Identification of source code lines is done by assigning a line number to each source line. These numbers are in the range of 0 through 167_8 and are assigned by DEMOT. User addressing within a module for JUMPS and FORMAT statements is accomplished through user-assigned statement numbers in the range of 0 through 77_8. A statement number may be defined only once within a module.

Registers and Buffers

Fixed names are assigned to registers and buffers used as variables within the DEMOT language. The following *programmable* registers exist in the PP driver.

Register	Function
B0-B15	Index registers for arithmetic and indexing.
WC	Word count register for I/O operations.
BA	Beginning address for buffer first word address (fwa) for I/O.

The following *read-only* registers exist as operands.

Register	Function
P0-P9	Parameter registers.
WT	Words transmitted on the last I/O operation.
LF	Last function code issued to I/O.
EC	Last error code posted.
EA	Error address; line where last error occurred.
EM	Error message index.
ES	Executive switches (stop on error, repeat module, and so on). These switches are used by DEMOT diagnostic modules which are executing in a PPU and are driving a device directly connected to the I/O channel.
DS	Diagnostic switches (repeat module, repeat condition and so on). These switches are used by DEMOT diagnostic modules which are executing remotely in a device such as the Loosely Coupled Network (LCN) or the Federal Standard Channel (FSC).
DC	Device code of the device being tested.
AL	Current access level to device assigned.
P	Program address register that identifies the current line number being executed within a module.
RT	Real-time PP clock. This value can be an operand only in a very few specific cases.

The following *programmable* buffers exist in the product overlays to the PP driver within the PP.

Buffer	Function
OB	Output buffer for writing data to devices.
IB	Input buffer for reading data from devices.
SB	Status buffer for input of device status.

Buffers entries can be referenced with ordinals, called subscripts of octal, decimal, or equated values; registers; or registers plus an octal or decimal value. Zero is the first word of any buffer.

Examples:

`IB (20)`	Octal ordinal (default).
`OB (8)`	Decimal ordinal.
`SB (GENST1)`	Equated ordinal.
`OB (B1)`	Register ordinal.
`SB (B1 + 5)`	Register plus value ordinal.

The occurrence of an 8 or 9 within any numeric field indicates that the field is a decimal-number field. If a postradix of D or B is used, the number is converted according to the radix.

Examples:

`LINE = 78`	The number 78 equals 116_8.
`LINE = 72D`	The line number equals 110_8.
`ASSIGN, AL=8, CH=1`	The AL field equals 10_8.

All references to a buffer imply all of the above ordinals. The compiler supports the following keywords as equated values to allow easy reference to specific bits of the ES or DS registers. These values may be used anywhere a constant or an equated value is legal.

ES Register equates keywords as follows:

Keyword	Description	Keyword	Description
RM	Repeat module.	SE	Stop on error.
RC	Repeat condition.	SL	Scoping loop (RM or RC, not SE).
MS	RM or SE.	CM	Repeat condition or module.
CS	RC or SE.	TM	Terminal mode.
DL	Dayfile log-on.	PL	Print log-on.
PD	PL or DL.		

DS Register equates keywords as follows:

Keyword	Description	Keyword	Description
RM	Repeat module.	SE	Stop on error.
RC	Repeat condition.	CM	Repeat condition or module.
MS	Repeat module or stop on error.	BC	Beginning of condition stop.
CS	Repeat condition or stop on error.	BT	Beginning of test stop.
BM	Beginning of module.	TE	Test end stop.
CE	Condition end.	ME	Module end stop.
S1	Programmable switch 1.	S2	Programmable switch 2.

Data Organization

Executive directives and source code lines may consist of a maximum of 60 decimal characters per line. Directives are decoded and acted upon immediately. Source code data is saved for later compilation in a fixed format of 60 decimal characters (six CM words) per line, and a maximum of 120 decimal source lines may exist within a module. Source code commands must be in a defined order to be interpreted by the compiler. The following conditions apply.

- Statement labeling within a module is by user-defined statement numbers of 0 through 77B or 0 through 63D that must appear in columns 1 and/or 2. Single-digit statement numbers may start in column 1 or 2, with a leading 0 optional. Statement numbers may be decimal numbers in the range of 0 through 63 if a BASE DECIMAL statement appears in the data division of the module.

- A source line is assumed to be 60 decimal characters long. Any information past column 60 is ignored by the compiler.

- Column 3 must be blank on noncomment statements, except for the DATA statement continuation cards in which a comma is used to indicate continuation. Comment statements are free field.

- Command keynames must start in column 4 or 5, and command parameters continue until two successive blanks are detected. Each command starts with a keyname followed by an expected sequence of parameters.
- Commas, periods, or single blanks are used to separate keynames and parameters. Two successive commas or periods cause an octal 0 to be stored as a parameter, and this is repeated for each subsequent comma or period following the first two.

  Examples:

DATA (OB(10), CON) 1,2,,3	Data items = 1, 2, 0, 3.
IB (10) = B1 + 1234	Two blanks terminate string.

- The plus (+), minus (–), left parenthesis ((), right parenthesis ()), equal sign (=), dollar sign ($), asterisk (*), and slash (/) are separators, but are also considered to be parameters themselves. This means that if one of these separators is included in the syntax of a command, it is not optional; it must be specified.
- Once a command string has been properly terminated with two successive blanks, the remaining characters, if any, may be used as comments for documentation purposes.

Operational Procedure

The DEMOT executive and driver are called and placed into execution by a series of CMSE commands. These commands can be entered at the keyboard.

To execute the command buffer, the operator enters:

```
GO,name
```

where the following parameter applies:

name	Name of the command buffer.

Refer to section 5 for typical command buffers to place the executive/driver and compiler into execution from cards.

All of the print log output is placed on an MSL file called OUTPUT. To dump this file, the MSL utility, DMP, must be used. The SETSW, PL directive must be given before running the diagnostic. The following CMSE command dumps this file.

```
CP,DMP
RU,/,100
PP,/
```

This brings up the parameter display for DMP. The F,OUTPUT command dumps the output file. If the message DUPLICATE FILE NAME-OUTPUT appears, an output file, OUTPUT, already exists and the new file is appended to the existing output.

Executive Directives

Table 5-1 provides a summary of DEMOT executive directives. These directives are read by the executive and acted upon immediately, if legal at the time of input. If illegal, an error message is written to the currently selected output files.

All executive directives may be preceded by a period, if desired, to distinguish them from CMSE commands.

Table 5-1. Executive Directives Summary

Directive	Description
ASSIGN or A	Assigns a device to DEMOT for testing.
CLRDS	Clears switch options within the DS register.
CLRSW	Clears switch options within DEMOT.
COMPILE or C	Calls the compiler to compile a module.
DEVICES	Displays list of device codes supported.
DEMOT	Displays DEMOT executive directives.
DEMOT C	Sets compiler to active or inactive.
DEMOT S = xxxxxx	Sets name of file to be used for SYS read.
DROP	Stops the module; idles EXEC/driver PPs.
GO or G	Resumes execution of a stopped module.
I or D	Increments or decrements to next display page.
LINE or L	Sets line number to receive next source.
LOAD	Loads module from library.
LOADCM	Loads specified modules into CM.
LOADCM S=xxxxxx	Loads scratch file for SYS read into CM.
MOVE or M	Opens up or eliminates source code lines.
PARAM or P	Inputs parameters to pass to PP registers.
PICTURE	Forms picture of DEMOT display on output.
PPU	Displays PP messages/plots.
RUN or R	Starts PP executing selected modules.
SCRATCH	Change record accessed by SYSRD/SYSREW/SYSRD.
SETDS	Sets switch options within the DS register.
SETSW	Sets switch options within DEMOT.
SOURCE	Displays source code of current module.
STOP or S	Stops execution of a module in the PP.

ASSIGN Directive

The ASSIGN directive assigns a device to DEMOT for testing. It uses one of the following formats:

```
ASSIGN CH=aa,EQ=bb,UN=cc,DC=dd,SN=eeeeee,AL=ff.
    or
A,CH=aa,EQ=bb,UN=cc,DC=dd,SN=eeeeee,AL=ff.
```

where the following parameters apply:

aa	Channel (default = 0).
bb	Equipment number (default = 0).
cc	Unit number (default = 0).
dd	Device code (default = 0).
eeeeeee	Media serial number (default = 0).
ff	Access level (default = 0).

NOTE

Be sure to provide an access level (AL) with every ASSIGN directive. If the access level is not provided, AL is set to zero.

For a mass storage system (MSS), the following format is used:

```
ASSIGN...,MST=gg, MSA=hh, MSTPH=ii, CSU=jj
```

where the following parameters apply:

gg	MST address (default = 0).
hh	MSA address (default = 0).
ii	MST physical address (default = 0).
jj	CSU address (default = 0).

For the LCN, the following format is used:

```
ASSIGN...,LTA=kkk,TCU=jj,AC=mmmm,BFZ=nn
```

where the following parameters apply:

kkk	Logical Trunk Control Unit (TCU) address, used if DC=301 (octal).
jj	Octal number of enabled TCU (1=TCU 3, 2=TCU 2, 4=TCU 1, 10=TCU 0). Used if DC=301 (octal).
mmmm	Access code of assigned Network Access Device (NAD), used if DC=301 (octal).
nn	Octal value representing controlware buffer size (1=516 bytes, 2=2064 bytes, 4=4128 bytes). Used if DC=301 (octal).

The ASSIGN directive provides the required data to access a device for testing. There are several formats of the ASSIGN directive to provide the required data to gain this access. Access levels 0 through 4 indicate that a high-level language and product overlay is to be used. If the controller has controlware memory, its contents cannot be changed.

Access level 5 indicates that only the assigned device may have its controlware destroyed. Access level 5 is used for LCN NAD tests. Access level 6 indicates that both the assigned and accessed devices may have their controlware destroyed. Access level 6 is for LCN NAD local and remote testing.

To gain access to the low-level I/O commands, access level 10, 20, or 21 must be specified.

Table 5-2 defines the access level codes and the level of testing that is allowed at each level.

Table 5-2. Access Level Codes

Level Octal	RMS	Non-RMS
0	Illegal (if specified on assign).	Dedicated unit, high-level I/O.
1	Read preallocated area only, high-level I/O.	Illegal.
2	Read any normal data area only, high-level I/O.	Illegal.
3	Read any normal data area, write preallocated area, high-level I/O.	Illegal.
4	Read/write any normal data area, high-level I/O dedicated unit.	Illegal.
5	Illegal.	LCN NAD dedicated. Code in NAD assigned may be rewritten.
6	Illegal.	LCN NAD local and remote dedicated. Code may be rewritten in both.
10	Dedicated controller and all units. High- and low-level I/O. May destroy controlware.	Dedicated controller and all units. High- and low-level I/O. May destroy controlware.
20	Dedicated controller, all units and channel. All devices must be off and not in use on the channel since master clears may be performed. High- and low-level I/O. Controlware may be destroyed.	Dedicated controller, all units and channel. All devices must be off and not in use on the channel since master clears may be performed. High- and low-level I/O. Controlware may be destroyed.
21	Same as level 20 plus I/O channel is left in a hung state if a channel hang occurs.	Same as level 20 plus I/O channel is left in a hung state if a channel hang occurs.

Prior to allowing any module to be executed, DEMOT verifies that the module requested was actually written for the device code that is currently assigned. A module may support as many as 10 unique device codes.

CLRDS Directive

The CLRDS directive sets switches logically off in the DS register. It uses the following format:

```
CLRDS aa,bb,...,ff
```

where parameters aa through ff may be any of the following:

Switch	Default	Bit	Description
SE	On	0	Stop on error.
RM	Off	1	Repeat module.
RC	Off	2	Repeat condition.
BC	Off	3	Beginning of condition stop.
BM	Off	4	Beginning of module stop.
BT	Off	5	Beginning of test stop.
TE	Off	6	End of test stop.
CE	Off	7	Condition end stop.
ME	Off	8	Module end stop.
S1	Off	9	Programmable switch 1.
RT	Off	10	Repeat test.
S2	Off	11	Programmable switch 2.

CLRSW Directive

The CLRSW directive clears switch options within DEMOT. It uses the following format:

```
CLRSW,aa,bb...,ff
```

where aa through ff indicate clear switch options as follows:

Switch	Default	Description
RC	Off	Repeat condition.
SE	On	Stop on error.
RM	Off	Repeat module.
PL	Off	Print log (off stops all new output).
DL	Off	Dayfile log (off stops all new dayfile messages).
RT	Off	Repeat test.
KL	Off	System display.

COMPILE Directive

The COMPILE directive calls the compiler to compile the source code buffer. It uses one of the following formats:

```
COMPILE a,b

    or

C a,b
```

where the following parameter applies:

a,b Compiler options as follows:

Option	Description
LIST	Causes the source and binary of the module compiled to be written to the output file if the print log (PL) switch is on.
SOURCE	Causes the source code to be carried with the module binary when it is saved on the library after being compiled.

DEMOT Directive

The DEMOT directive displays the executive directives (figures 5-3 and 5-4). It uses the following format.

```
DEMOT
```

```
DIRECTIVE                    COMMENTS

  ASSIGN      ASSIGN DEVICE FOR DEMOT USE *ASSIGN=AA
              EQ=BB, UN=CC, DC=DD, SN=EEEEEE, AL=FF*
  CLRDS       CLEAR DS SWITCHES *CLRSW SE,RT,DL,RM,PL,KL,RC*
  COMPILE     COMPILE SOURCE BUFFER *COMPILE,SOURCE,LIST*
  DEVICES     DISPLAY DEVICE CODES *DEVICES*
  DROP        DROP DEMOT PPUS *DROP*
  GO          RESUME EXECUTION OF MODULE THAT WAS STOPPED *G*
  LINE        SET LINE FOR NEXT ENTRY *LINE*
  LOAD        LOAD MODULE *LOAD MNENN*
  LOADCM      LOAD MODULES TO CM *LOADCM MNE (00,01,02,03,03A)*
              OR *LOADCM S=SYSFIL*
  DEMOT       DISPLAY DEMOT EXECUTIVE DIRECTIVES *DEMOT C=A/I*
              OR CHANGE SYS FILE NAME *DEMOT S=SYSFILE*
  MOVE        *MOVE AAA UP BBB LINES*
  MOVE        *MOVE AAA DOWN BBB LINES*
```

Figure 5-3. Sample Display for DEMOT Directive (First Page)

```
DIRECTIVE                    COMMENTS

  PARAM       CHANGES PARAMETERS *PARAM P0=1234,P1=1234,P9=1234
              LL=1234 LL=LINE LIMIT
  PICTURE     PRINT DISPLAY ON OUTPUT FILE *PICTURE=TYPE*
  PPU         DISPLAY PPU MESSAGES/PLOTS *PPU*
  RUN         RUN MODULE(S) *RUN MNE (10,02,03,04,10,07)*
  SCRATCH     CHANGE SYSRD RECORD NAME *SCRATCH=LFN*
  SETDS       SET DS SWITCHES *SETDS,RC,RM,SE,TE,DE,RT
  SETSW       SET SWITCHES *SETSW,SE,RC,RM,DL,PL,KL,RT*
  SOURCE      DISPLAY SOURCE FOR MODULE *SOURCE*
  STOP        STOP PPU EXECUTION *STOP*
  I           INCREMENT PAGE ON DISPLAY *I*
  D           DECREMENT PAGE ON DISPLAY *D*
 .X-X         ENTER SOURCE LINE FROM KEYBOARD *.SOURCE*
```

Figure 5-4. Sample Display for DEMOT Directive (Second Page)

DEMOT C Directive

The DEMOT C directive sets the compiler to active or inactive status. It uses the following format:

```
DEMOT C= a
```

where the following parameter applies:

a   Status of compiler as follows:

Code	Status
A	Active
I	Inactive

DEMOT S Directive

The DEMOT S directive changes the name of the scratch file used with the SYS command. The default name is SCRATCH. You must enter this command before MLD is initiated. The DEMOT S directive uses the following format:

```
DEMOT S=xxxxxx
```

where the following parameter applies:

xxxxxx   Name of the scratch file.

DEVICES Directive

The DEVICES directive displays device codes for use on the ASSIGN directive (figures 5-5, 5-6, and 5-7). It uses the following format.

```
DEVICES
```

```
                        CODES FOR DEVICES

01=841                          02=7054/844-2 DISK
03=7054/844-4 DISK              04=7154/844-2 DISK
05=7154/7155/844-4 DISK         06=819 DISK
07=7155/885 DISK                10=FSC-100 MBYTE
11=FSC-200 MBYTE                12=FSC-317 MBYTE
13=RESERVED                     14=7155/885-42 DISK (DEMA)
15=RESERVED                     16=7155/885 DISK (LSFMD)
17=RESERVED                     20=405 CARD READER
21=415 CARD READER              22=512 LINE PRINTER
23=580-12 LINE PRINTER          24=580-16 LINE PRINTER
25=580-20 LINE PRINTER          26=580-12 PFC LINE PRINTER
27=580-16 PFC LINE PRINTER      30=580-20 PFC LINE PRINTER
```

Figure 5-5. Sample Display for DEVICES Directive (First Page)

```
                        CODES FOR DEVICES

31=CCC/5870 PRINT (NIP)         32=CCC/5970 PRINTER (NIP)
33-37=RESERVED                  40=60X 7 TRK TAPE
41=65X 7 TRK TAPE               42=66X 7 TRK TAPE
43=RESERVED                     44=67X 7 TRK TAPE
45=FSC 7 TRK TAPE               46=RESERVED
47=CCC/5680 CARTRIDGE TAPE      50=60X 9 TRK TAPE
51=65X 9 TRK TAPE               52=66X 9 TRK TAPE
53=698 9 TRK TAPE               54=67X 9 TRK TAPE
55=67X GCR 9 TRK TAPE           56=FSC 9 TRK TAPE
57=639 9 TRK TAPE               60=6671 MUX
61=6676 MUX                     62=2250-100 EMULATING 6671
63=2550-100 EMULATING 6676      64=2550
65=7077-1 LCC                   66=6673 DATA SET
67=6683 COUPLER                 70=DDP
71=ECS COUPLER                  72=RESERVED
```

Figure 5-6. Sample Display for DEVICES Directive (Second Page)

```
                    CODES FOR DEVICES

73-77=RESERVED                  100=MSS-MST
101=MSS-CSU                     102-103=RESERVED
104=CCC/MASSTOR SUBSYSTEM       105=CCC/MASSTOR (HPO)
110=7255/834 DISK (FSD I)       111=7255/836 DISK (FSD II)
115=CCC/895 DISK                171=ESM II SUBSYSTEM
200=CYBERPLUS RING PORT         211=STORNET SUBSYSTEM
300=LCN LOCAL NAD               301=LCN NAD REMOTE
302-377=RESERVED                400=CIU OUTPUT CONTRL
401=CIU INPUT CONTRL            402-477=RESERVED
500=CYBER/UNIBUS INTERFACE      501-7677=RESERVED
7700-7777=QSE

LINE = 0000    PP STATUS = STOPPED    MODULE = xxx00
```

Figure 5-7. Sample Display for DEVICES Directive (Third Page)

DROP Directive

The DROP directive stops the PP driver, if running, and drops DEMOT. It uses the following format.

```
DROP
```

GO Directive

The GO directive resumes execution of the PP driver that has stopped. It uses one of the following formats.

```
GO

  or

G
```

I or D Directive

The I or D directive increments or decrements, respectively, to the next display page. It uses one of the following formats.

```
I = a

  or

D = a
```

where the following parameter applies:

a	Advances or moves back from the current page being displayed on the system, the number of pages specified by a.

LINE Directive

The LINE directive sets the data input pointer to a specific source line number. The next source code input then is sent to this line number. It uses one of the following formats:

```
LINE a

    or

L a
```

where the following parameter applies:

a	Line number of 0 to 167_8.

LOAD Directive

The LOAD directive loads a module from the CMSE file to the module buffer and displays the first page of the source, if available. It uses the following format:

```
LOAD aaannn
```

where the following parameters apply:

aaa	Module test series.
nnn	Module number.

LOADCM Directive

The LOADCM directive loads specified modules into CM when running DEMOT from a tape system. Processing of this directive terminates with an error message if a specified module cannot be found on the library. The LOADCM directive uses the following format:

```
LOADCM aaa(bbb,cc,ddd)
```

where the following parameters apply:

aaa	Module test series.
bbb through ddd	Two- or three-character module or submodule numbers.

LOADCM S Directive

The LOADCM S directive loads specified scratch files into CM, after the modules have been loaded. It uses the following format:

```
LOADCM S=xxxxxx
```

where the following parameter applies:

xxxxxx	Name of the scratch file to be used by the SYS read.

MOVE Directive

The MOVE directive moves source lines up or down. It is used to make space for modifications to a module or to eliminate unwanted lines. A move up overwrites the data currently existing in the lines being moved to. It uses one of the following formats:

```
MOVE a UP b     or   M a UP b   or   MOVE a + b   or   M a + b

MOVE a DOWN b   or   a DOWN b   or   MOVE a - b   or   M a - b
```

where the following parameters apply:

a	Line number to start the move.
b	Number of lines to move.

PARAM Directive

The PARAM directive passes program parameters to the PP driver when a module is started into execution. The display is changed to the PARAM display (figures 5-8 and 5-9). The PARAM directive uses one of the following formats:

```
PARAM aa=xxxx, aa=xxxx,...

    or

P aa=xxxx, aa=xxxx,...
```

where the following parameters apply:

aa — Indicates one of the following:

Code	Description
P0 to P9	PP driver register number.
LL	The number of lines allowed (line limit) before DEMOT turns print output off. (LL set to 0 = infinity.)

xxxx — Octal value to be placed into the specified PP driver register.

```
                    PARAMETERS IN OCTAL

    P0=0000   P1=0000   P2=0000   P3=0000   P4=0000   P5=0000
    P6=0000   P7=0000   P8=0000   P0=0000             LL=1750
 ____________________________________________________________

                    ES SWITCH STATUS

     ON  - (SE) STOP ON ERROR         OFF - (RM) REPEAT MODULE
     OFF - (RT) REPEAT TEST
     OFF - (PL) PRINT LOG             OFF - (DL) DAYFILE LOG
     OFF - (LK) K/L DISPLAY           OFF - (RC) REPEAT CONDITION
 ____________________________________________________________

                       ASSIGN VALUES

         DC=0000   CH=0000   EQ=0000   UN=0000   AL=0000
         SN=000000

 ____________________________________________________________

                     SCRATCH = SCRATC
```

Figure 5-8. PARAM Directive with Default Values Sample Display (Page 1)

```
                    DS SWITCH STATUS

     ON  - (SE) STOP ON ERROR         OFF - (BT) BEGINNING OF TEST
     OFF - (RT) REPEAT TEST           OFF - (TE) TEST END
     OFF - (RM) REPEAT MODULE         OFF - (CE) CONDITION END
     OFF - (RC) REPEAT CONDITION      OFF - (ME) MODULE END
     OFF - (BC) BEGINNING OF COND     OFF - (S1) PROG SWITCH 1
     OFF - (BM) BEGINNING OF MODULE   OFF - (S2) PROG SWITCH 2

                ADDITIONAL VALUES FOR MSS ONLY

           CSU=0000   MSA=0000   MST=0000   MSTPH=0000

             ADDITIONAL ASSIGN VALUES FOR LCN ONLY

           LTA=0000   TCU=0000   AC=0000   BFZ=0000

         LINE=000   PPU STATUS=STOPPED   MODULE=xxx00
```

Figure 5-9. PARAM Directive with Default Values Sample Display (Page 2)

PICTURE Directive

The PICTURE directive is used to print a picture of a specified display. It uses the following format:

```
PICTURE type
```

where the following parameter applies:

type — Specifies display to be printed as follows:

Code	Prints
DEMOT	DEMOT directives display.
DEVICES	Devices display.
PARAM	Parameter display.
PPU	PP display.
SOURCE	List one page of source data. To obtain a complete source data listing, use the COMPILE directive with the LIST option (*COMPILE,LIST,SOURCE*)
Blank	Current display.

PPU Directive

The PPU directive displays module-generated information. It uses the following format.

```
PPU
```

RUN Directive

The RUN directive starts PP execution. It uses one of the following formats:

```
RUN sss (aa,bb,cc,dd,ee,ff,gg,...)

  or

R sss (aa,bb,cc,dd,ee,ff,gg,...)
```

where the following parameters apply:

sss — A three-alphanumeric-character test series name.

aa,bb, and so on — Decimal module numbers within the specified series to be executed in the order defined. (If you specify only the test series, all modules in the series are executed in numerical order up to the first END command.)

Leading zeros are necessary when specifying module numbers. The RUN directive specifies the test series of modules that are to be executed, the specific modules in this series to execute, and the order in which to execute them. If more than 60 characters are needed to specify module numbers, do not include the right parenthesis character until the last line of the directive. As many as 100 modules may be specified on one RUN directive.

Examples:

1. Run specific modules regardless of END/EXIT statements.

   RUN TST (00,01,02,03,04,05,06,07....29,30,31,32,33,34)

   Runs modules 00 through 34 in order specified.

   RUN TST (00,01,00,01)

   Runs modules 00,01,00,01.

2. Run a test series.

RUN TST	Runs test series starting with 00.
RUN TST20	Runs test series starting with 20.
RUN TST30	Runs test series starting with 30.

NOTE

A false CMSE error message `ERR-NOT ON LIBRARY`, may be encountered if a break exists in the numerical series. Execution is not affected.

3. Run single module.

RUN TST (00)	Runs module 00.
RUN TST (01)	Runs module 01.
RUN	Runs currently loaded module.

SCRATCH Directive

The SCRATCH directive changes the record name accessed by SYSRD/SYSREW/SYSWR to lfn. It uses the following format:

```
SCRATCH=lfn
```

where lfn is from 1 to 6 characters.

SETDS Directive

The SETDS directive sets switches logically on in the DS register and uses the format:

```
SETDS aa,bb,...,ff
```

where parameters aa through ff may be any of the following:

Default	Switch	Bit	Description
On	SE	0	Stop on error.
Off	RM	1	Repeat module.
Off	RC	2	Repeat condition.
Off	BC	3	Beginning of condition stop.
Off	BM	4	Beginning of module stop.
Off	BT	5	Beginning of test stop.
Off	TE	6	End of test stop.
Off	CE	7	Condition end stop.
Off	ME	8	Module end stop.
Off	S1	9	Programmable switch 1.
Off	RT	10	Repeat test.
Off	S2	11	Programmable switch 2.

SETSW Directive

The SETSW directive turns on switch options and sends them to the PP driver. It uses the following format:

```
SETSW aa,bb,cc,...
```

where the following parameters apply:

aa,bb,cc, and so on — Switch options to be set as follows:

Switch	Default	Description
RC	Off	Repeat condition.
SE	On	Stop on error.
RM	Off	Repeat module.
PL	Off	Print log (On resumes new output).
DL	Off	Dayfile log (On resumes new dayfile messages).
RT	Off	Repeat test.
KL	Off	System display used for input.

NOTE

PL and DL must remain OFF when running DEMOT from tape.

SOURCE Directive

The SOURCE directive displays a page of the source code buffer (figure 5-10). The I directive can be used to advance the page. The LINE=0 directive can be used to reset the page to the beginning. It uses the following format.

```
SOURCE
```

```
SOURCE

        000=     MODULE TST01, 3000 (20, 21, 22)
        001=01   FORMAT SAMPLE MODULE TO DISPLAY MSGS ON LINE=*OCT
        002=02   B2=0
        003=03   B1=0
        004=4    MSG 1 (B1) TO LINE B1
        005=     GOTO 4 WHILE (B1+1.LT.50B)
        006=     GOTO 3 WHILE (B2+1.LT.100B)
        007=     END 2
        010=
        011=
        012=
        013=
        014=
        015=
        016=
        017=
            :
        021=



        LINE=010    PPU STATUS=    STOPPED    MODULE = xxx00
```

Figure 5-10. Sample Display for SOURCE Directive

STOP Directive

The STOP directive flags the PP driver to stop the current module being executed at the next logical breakpoint. It uses one of the following formats.

```
STOP

   or

S
```

Messages

In the DEMOT system, messages can originate from either the executive or the compiler.

Executive Messages

The following is a list of messages issued by the DEMOT executive.

Message	Definition
ACCESS LEVEL NOT HIGH ENOUGH.	Module is low level. Access level for low level must be greater than or equal to 10.
ACTIVE MODULE NOT STOPPED.	A GO was issued to a module that was not stopped.
CMSE REJ 21 CALL.	CMSE rejected the call by the executive to read a module.
COMPILER NOT ACTIVE.	The compiler is not active. Can be activated with DEMOT C=A directive.
COMPILER NOT RESPONDING.	The executive did not receive a response from the compiler on a compile call.
DAYFILE LIMIT REACHED.	Dayfile entry limit reached. Dayfile log is turned off.
DIRECTIVE PARAMETER ERROR.	Parameter was in error. Reenter with correct parameter.
DISK LOAD ERR.	Executive received a disk error while trying to load a module.
DRIVER DETECTED ERROR, SEE DAYFILE.	The driver detected an error on initialization and terminated.
END OF TEST.	Test series being run has encountered an END statement.
FORMAT WAS OUT OF RANGE.	Format statement could not be found. Check module source for error.
ILLEGAL WHEN MODULE RUNNING.	Directive is illegal when module is running.
ILLEGAL COMPILER CALL.	Call received from compiler is illegal.
LANG CHANGE ON ACTIVE CALL.	A call was active and a module was requested that was not written in the same language of the module that performed the CALL or EXIT TO.
LINE LIMIT REACHED.	Print log line limit has been reached. Print log is turned off.
LINE VALUE OUT OF RANGE.	Value too large. Reenter with a value within range.

Message	Definition
LOAD MODULES BEFORE SYSFILE.	Modules were not loaded to CM prior to a request to load SYSFILE to CM.
MODULE DOES NOT SUPPORT DEVICE CODE.	Device codes specified in the MODULE command do not include currently assigned device code.
MODULE LENGTH ERR.	Module read by the executive was too long.
MODULE NOT LOADED.	The module called has not been loaded into CM or a module could not be found when processing a LOADCM directive.
MODULE NOT RUNNING.	Directive is illegal when module is not running. (Example: STOP when module not running.)
MODULE NOT WRITTEN - DUPLICATE NAME = name.	An attempt was made to recompile a module already on the MSL library.
MODULE WRITTEN = name.	Module was compiled correctly and binary placed on the library.
MULT CALLS.	A CALL command was executed while a previous call was still active.
NO DEVICE ASSIGNED.	Device is not assigned. Enter ASSIGN directive before trying to run module (with device code nonzero).
OVERLAY LOAD ERR.	The executive received an error indication while trying to read an overlay from CMSE.
RUN DIRECTIVE NEEDS).	RUN directive was entered and never terminated. Enter a right parenthesis to terminate run directive.
SOURCE BUFFER FULL.	More than 120 source statements were entered. No more source entries are accepted until an executive directive is encountered.
SYNTAX ERROR. DIR = ...	Syntax error on input directive. Reenter corrected directive.
TOO MANY MODULES ON RUN DIRECTIVE.	RUN directive can contain only 100 module numbers.
UNKNOWN MODULE.	Module name not found on MSL file.
UNKNOWN MODULE SERIES.	The module series is not on the MSL library.
VERSION NUMBER ERR.	Current version of DEMOT does not match version number stored in module. If the user wants to run module, it must be recompiled.

Compiler Messages

The following is a list of messages posted by the DEMOT compiler.

Message	Definition
BUFFER SUBSCRIPT ILLEGAL.	Entry ordinal is illegal for IB, OB, or SB.
COMMENT CARD OUT OF MODULE/END CARD RANGE.	A / comment card was detected prior to the first MODULE card or following an END card.
COMPILER ERRORS - MODULE = name.	Module had compiler errors.
DOUBLE-DEFINED STATEMENT NUMBER.	Two statements have the same value assigned to them.
EQUATE TABLE FULL - MAXIMUM OF 20 ALLOWED.	Maximum number of equates to use is only 20.
EQUATED NAME HAS AN ILLEGAL LENGTH.	Equated name too long.
FORMAT BLOCK FULL - MAXIMUM OF 24 ALLOWED	Maximum number of formats is 24 octal. Reduce number of formats to assemble module.
ILLEGAL FORMAT STATEMENT NUMBER.	Format statement number is illegal.
MODULE CARD ERROR MODULE = name.	A legal module card was not found.
MODULE OVERFLOW - REDUCE SIZE.	Binary of module does not fit within bounds of module buffer. Alter source to correct overflow.
NO END STATEMENT - MODULE = name.	END statement not encountered within module source.
OCTAL OR EQUATED VALUE IS OUT OF RANGE.	Octal or equated value too large or too small.
SOURCE OVERFLOW MODULE = name.	More than 167_8 source lines were found.
STATEMENT OVERFLOW REDUCE SIZE.	An IF statement generated too much code. Reduce complexity of the statement.
SUBSCRIPT IS NOT TERMINATED.	Subscript must be terminated with a right parenthesis).
SYNTAX ERROR - COLUMN 3 MUST BE A BLANK.	Column 3 must be blank except on continuation lines.

Message	Definition
SYNTAX ERROR - STATEMENT NUMBER MUST BE 0-77B.	Value used on statement number is too large.
SYNTAX ERROR.	Syntax of statement is in error.
UNDEFINED STATEMENT NUMBER.	Statement number not found.
UNRECOGNIZED STATEMENT.	Statement is unrecognized by the compiler.

Typical DEMOT Jobs

Running DEMOT may be divided into three types of operations.

- Getting the executive and the drivers up and running. This must precede any other DEMOT operations.
- Calling the compiler to compile source code into binary. This operation is optional.
- Running of binary modules of a diagnostic. This may be done on a binary module just compiled or on binary modules saved from a previous compilation. (An example of typical source code is also shown to illustrate its appearance.)

Getting DEMOT Up and Running

Before anything else can occur, DEMOT must be up and running. Up and running means that the executive and the drivers are loaded and executing.

DEMOT always runs under CMSE. If CMSE is running on disk, use the following command buffers to load and start executing the executive and drivers.

DEMOT	To be loaded and executed in NIO PPs.
DEMCIOL	To be loaded and executed in CIO PPs, lower cluster of an AT478/AT481 IOU.
DEMCIOU	To be loaded and executed in CIO PPs, upper cluster of an AT478/AT481 IOU.
DEMI4C0	To be loaded and executed in cluster 0 of an AT511/AT512 IOU.
DEMI4C1	To be loaded and executed in cluster 1 of an AT511/AT512 IOU.
DEMI4C2	To be loaded and executed in cluster 2 of an AT511/AT512 IOU.
DEMI4C3	To be loaded and executed in cluster 3 of an AT511/AT512 IOU.

NOTE

The command buffers are to be used as examples and PPs/channels may have to be modified for certain configurations. The following command buffer is an example of how to load and execute DEMOT across cluster boundaries.

```
$$DEMI4CS
.*************************************************
.*  THIS COMMAND BUFFER IS AN EXAMPLE OF HOW TO    *
.*  LOAD AND EXECUTE DEMOT IN TWO CLUSTERS; NAMELY *
.*  CLUSTER 0 AND CLUSTER 1.  USES PPS 4 AND 5.    *
.*  CHANNEL 13 IS USED FOR THE COMMUNICATION       *
.*  CHANNEL BETWEEN MCX AND MLD.                   *
.*************************************************
LT,C04,MCX,,13,, ASSIGN MASTER PP 4 + COMM CH 13
CP,C05,MLD,,13,, ASSIGN SLAVE  PP 5 + COMM CH 13
SQ,77,1,C4
DEMOT,S=SCRATCH
RU,C05,100
TB
```

DEMOT is now ready to process executive directives to execute modules. CM is used for display buffers.

Two error messages may appear. The message ERR-NOT ON LIBRARY appears any time a broken series of modules is run. The message DUPLICATE FILE NAME-OUTPUT appears if the file OUTPUT has already been put in the library. Any further output is appended to the original file OUTPUT.

Using the DEMOT Compiler From Disk

The DEMOT compiler creates binary modules from source code. The compiler must be loaded and running in order for documentation on the source code to be examined.

NOTE

Run command buffer DEMOT and load microcode before executing command buffer DEMOTCx.

Run the appropriate DEMOTCx command buffer, as follows, to activate the DEMOT compiler.

DEMOTC1	With CIP 810/815 and CIP 825/830.
DEMOTC2	With CIP 835.
DEMOTC3	With CIP 840, CIP 845/850, and CIP 855/860/870.
DEMOTC3	With CIP 960 and CIP 962.
DEMOTC5	With CIP 990/995, CIP 992, and CIP 994.

NOTE

The DEMOT complier cannot be executed from a secondary IOU on dual IOU systems.

Example 1:

Run DEMOT from the tape library (DEMOT overlays in CM).

This sequence loads executive and driver overlays to CM as a part of initialization. This sequence can also be used with disk-based libraries.

The following is a program example.

Command	Definition
DP4	Deadstart PP 4.
DP5	Deadstart PP 5.
TL4,MCX,,5	Load and run MCX (executive) into PP 4; use channel 5 for communication.
TL5,MLD,,5	Load and run MLD (driver) into PP 5; use channel 5 for communication.
PP4	Select executive display.

DEMOT is now ready to process executive directives to execute modules. CM is used for overlay storage and display buffers.

NOTE

If a scratch file is required for testing, the LOADCM command should be used to load test modules and/or scratch file data to CM.

Following is the recommended procedure for running DEMOT disk tests on a system with only one channel access to disk.

NOTE

Run DEMOT from the tape library.

1. Examine ordinals 6, 7, and 8 of the Initial CMSE Display to ensure that the channel to be tested is not assigned to CMSE.
2. Set USE CM ordinal of the Initial CMSE Display in order to use the CM.
3. Load CMSE.
4. Run DEMOT from the tape library using the command sequence described above.

The tape loader, the CMSE monitor, and DEMOT all use CM. The only reference to tape required from this point on is to read the diagnostic modules. Since these modules are in sequential order on the tape, you should use the RUN,xxx command without specifying module numbers to run the diagnostic.

Total diagnostic execution time should be approximately equal to disk mode.

Using the DEMOT Compiler From Tape

The DEMOT compiler creates binary modules from source code. The compiler must be loaded and running in order for documentation on the source code to be examined.

Example 2:

Run DEMOT with compiler (using CM).

Before this command sequence can be executed, the DEMOT executive and drivers must have been initialized as shown in example 1.

NOTE

Load microcode and put the CPU in CYBER 170 state before executing this command sequence.

The following is a program example.

Command	Definition
HC	Halt CPU.
CC,MLC	Load MLC (compiler).
DP,n	Deadstart PP n.
EPn,100,2001	
EPn,101,0000	
EPn,102,2600	Enter values into PP n memory.
EPn,103,0100	
EPn,104,7606	
RUn,100	Run driver at address 100.
CN,1016	Define CMSE/CPU interface block.
DEMOT C=A	Set compiler active.

DEMOT is now ready to process LOAD and COMPILE directives. The LOAD directive can be used to examine the source code carried with modules on the CMSE library (xxx99 modules contain user documentation as part of their source). The COMPILE directive can be used to modify or create modules if running from a disk library.

Running DEMOT Diagnostics

These two remaining command sequence examples show how binary modules may be executed.

Example 3:

Run DEMOT diagnostic from disk library.

Before this command sequence can be executed, the DEMOT executive and drivers must have been initialized using command buffer DEMOT. Initialization of the compiler as shown in example 1 is optional.

The following is a program example.

Command	Definition
ASSIGN CH=10,EQ=2,UN=3,DC=40	Assign the required channel, equipment number, unit number, and device type.
PARAM P0=100,P1=1234	Preset the required parameter registers.
SETSW SE,PL	Set control switches.
RUN T7X(00,15,21)	Run selected test modules. In this example, modules 00, 15, and 21 of diagnostic T7X are to be run in that order.

To run all modules of T7X, the last command is changed to read:

Command	Definition
RUN T7X	Run entire default diagnostic.

Example 4:

Run DEMOT diagnostic from tape library.

Before this command sequence can be executed, the DEMOT executive and drivers must have been initialized using command buffer DEMOT. Initialization of the compiler as shown in example 1 is optional.

The following is a program example.

Command	Definition
ASSIGN CH=10,EQ=2,UN=3,DC=40	Assign the required channel, equipment number, unit number, and device type.
PARAM P0=100,P1=1234	Preset the required parameter registers.
LOADCM T7X(00,15,21)	Load selected modules to CM.
LOADCM s=xxxxxx	Load scratch file xxxxxx to CM. This command is necessary only if the diagnostic to be run accesses scratch file data.
RUN T7X(00,15,21)	Run selected test modules. In this example, modules 00, 15, and 21 of diagnostic T7X are to be run in that order.

Example 5:

The following is the source code for a typical module written to execute on a 67X tape subsystem. This source code could be compiled into executable PP code by the DEMOT compiler and executed by the PP driver and 66X product overlay.

```
     MODULE T7X01,67X(54)
/
/    THIS IS AN END-OF-FILE TEST FOR A 67X ATS.
/    THE TAPE IS REWOUND, 200 OCTAL TAPE MARKS WRITTEN,
/    AND THE PROCESS REPEATED FOR 10 OCTAL TIMES.
/    EOF STATUS IS TESTED AFTER EACH ENDFILE.
/    TERMINATE TESTING AFTER 50 OCTAL ERRORS.
/
10    FORMAT NO EOF ON LOOP *OCT RECORD *OCT,STATUS= *OCT
      EQUATE EOF TO 20
1     B1 = 0
      B3 = 0
2     B2 = 0
      RES P1
      REWIND
3     RES P1
      ENDFILE
      IF( SB(1) AND EOF EQ EOF) GOTO 4                      IF NO ERROR
      MSG 10(B1,B2,SB(1)) TO LINE B3
      GOTO 4 WHILE (B3 + 1 NE 50)                           IF BUFFER NOT FULL
      PICTURE
      EXIT                                                  TERMINATE ON 40 ERRORS
4     GOTO 3 WHILE (B2 + 1 NE 200)
      GOTO 2 WHILE (B1 + 1 NE 10)
      EXIT
      END 1
```

Utility Programs 6

Utility Programs 6

Utility programs provide capabilities not covered under standard common maintenance software executive (CMSE) commands or tests.

There are two types of utilities. First are those that are part of CMSE. These utilities are usually overlays executed by a CMSE command. The other type are the standalone utilities, which may require a supporting system to perform their function, but are maintained as a separate program. These programs usually have their own user interface commands. Both types of utilities are described in this section.

Real-Time Display and Timing Clock Program (CLK)

The real-time display and timing clock program (CLK) provides a digital real-time display that computes run times of diagnostic software and related hardware operations.

CLK samples the real-time clock channel about once every 45 milliseconds. After 1000 milliseconds elapse, the second counter increments. This same procedure is used for minutes and hours. After 24 hours, the clock returns to zero.

The time kept by CLK is inaccurate when peripheral processor (PP) memory is displayed. To ensure the correct time is kept when using CLK, the PP memory should not be displayed.

Hardware Requirements

CLK requires the following hardware.

- One PP
- One real-time clock with a 4.096-millisecond clock period
- Equipment required to load any diagnostic

Software Requirements

This program executes under the maintenance system.

Loading Procedure

This program is called into PP xx as CLK under the maintenance system with the command

```
LTxx,CLK,,p.
```

If the load parameter, p, is nonzero, an RTC test is run during initialization. If the load parameter is zero, no RTC test is run.

Parameters and Displays

If you type hhmmss. and press **(CR)** under the PP display, the time display is changed to:

```
TIME. hh.mm.ss.
```

hh	Hours
mm	Minutes
ss	Seconds

If you type r. and press **(CR)** under the PP display, the time is reset to zero.

Error Messages

```
ERROR-RTC NEVER GOES TO ZERO
```

RTC was never found to be zero.

```
ERROR-RTC EXP=xxxx REC=yyyy
```

RTC was expected to be xxxx, but was found to be yyyy.

Dayfile and Memory Dump Utility (DMP) (not applicable to CYBER 962/992)

The Dayfile and Memory Dump (DMP) routine permits dumping of the PPs, CM, and EM to a line printer without the need to deadstart the HDP routine. The DMP routine also dumps the dayfile. DMP operates under CMSE.

Hardware Requirements

DMP requires the following hardware.

- One PP
- One 512 or 580 line printer
- One library loading device

Software Requirements

DMP loads and runs under CMSE. The line printer train image memory and the format control memory (if applicable) must be loaded correctly prior to executing any dumps using DMP.

Loading Procedures

The DMP routine is called into a PP using the following CMSE command:

```
LT,DMP,,ccee
```

where the following parameters apply:

cc	Line printer channel number.
ee	Line printer equipment number.

An auto snap feature of DMP is initiated using the following CMSE command:

```
CP,DMP,,ccee,n
RU,/,100
```

where the following parameters apply:

cc	Line printer channel number.
ee	Line printer equipment number.
n	Dump to display. 1 = A display 2 = B display

Parameters

All parameter entries are displayed. The parameters may be changed at the end of any dump.

Messages

If a hang condition occurs because of a paper-out or not-ready condition, press the **Space Bar** to continue (after condition is corrected).

The following message appears while dumping central memory (CM).

```
DUMPING CENTRAL MEMORY
```

The following message appears while dumping PPxx.

```
DUMPING PPxx
```

The following message appears while dumping EM.

```
DUMPING EXTENDED MEMORY
```

The following message appears when PPxx is not communicating with the monitor. DMP requires a restart.

```
PPxx CANNOT BE DUMPED
```

The following message appears when the line printer is not ready. Check for a paper-out condition. After making the line printer ready, press the **Space Bar** to continue.

```
LP NOT READY
```

The following message occurs during a PFC load when a 6/8 line coincidence cannot be achieved after five single spaces of the line printer.

```
UNABLE TO GET COINCIDENCE - TYPE R
```

The following message occurs at the conclusion of a successful PFC memory load. After ensuring that the paper is at the top of the form, pressing the **Space Bar** clears the message.

```
PFC LOADED - SET PAPER TO TOP OF FORM
```

The following message occurs during an image memory load when an illegal printer type was previously selected. After changing address 1012, pressing the **Space Bar** clears the message.

```
ILLEGAL PRINTER TYPE - RESET 1012
```

The following message occurs during an image memory load when an illegal image type was previously selected. After changing address 1013, pressing the **Space Bar** clears the message.

```
ILLEGAL TRAIN TYPE - RESET 1013
```

The following message appears at the beginning of a dump. The user selects the type of dump desired from the option listed.

```
TYPE PXX, PNXX OR PCXX FOR PP DUMP
    PN(NIO) OR PC(CIO) FOR 8K
FOR 176-PPU(1-15), PP(21-31,40-51)
FOR 17X-PP(1-11,20-31)
DUMP PP MUST BE IDLED
NO SYSTEM PP MAY BE DUMPED
TYPE A - TO DUMP A DISPLAY
TYPE B - TO DUMP B DISPLAY
TYPE CX,Y. TO DUMP CM X TO Y
TYPE CX,Y,Z. TO DUMP CM X TO Y,RA=Z
TYPE I - TO LOAD IMAGE MEMORY
TYPE L - TO LOAD PFC MEMORY (580)
TYPE D - TO END DMP
TYPE EX,Y. TO DUMP EM X TO Y
TYPE EX,Y,Z. TO DUMP EM X TO Y,RA=Z
TYPE F - TO DUMP DAYFILE
TYPE F,NAME - TO DUMP FILE NAME
TYPE R - TO RESTART
1002 = LP(CCEE)
1007 = LINES/INCH (10=8LPI, 11=6LPI)
1012 = PRINTER (0=501, 1=512, 2=580)
1013 = TRAIN (1, 2, 3, 4, 5 OR 6)
```

The following message appears during a dump of the dayfile.

```
DUMPING DAYFILE
```

The following message appears while dumping the A or B display.

```
DUMPING DISPLAY
```

The following message appears during a dump to describe the dump from the address defined by F-F to the address defined by L-L, relative to the address defined by R-R.

```
TYPE CF-F, L-L, R-R. TO DUMP F TO L RELATIVE TO R
```

Tape-To-Disk Utility (TDX) (not applicable to CYBER 962/992)

The tape-to-disk utility (TDX) is a PP routine that enables the operator to build an 834, 836, 844, 885, or 895 disk library. TDX consists of two records: TDX and TD0. TDX is a disk driver that loads into PP1, and TD0 is a tape driver that loads into PP0. TDX operates in manual mode or in automatic mode.

TDX is not applicable to CYBER 962/992 systems.[1] TDX is capable of performing the following operations in manual mode.

- Building MSL on disk from tape
- Building a command buffer library on disk from tape
- Adding or replacing programs and command buffers to MSL on disk
- Copying programs and command buffers to tape from MSL on disk
- Displaying MSL tables

TDX operates in automatic mode when it is used to install the MSL portion of the CYBER Initialization Package (CIP) tape. In automatic mode, no user input is required throughout the installation as TDX is loaded and is supplied with preset tape, disk, and build type parameters.

TDX is capable of performing the following operations in automatic mode.

- Initial CIP tape installation
- Update CIP tape installation

For additional information about CIP automatic installation, refer to the appropriate CIP reference manual.

Hardware Requirements

TDX requires the following hardware for proper execution.

- Three active PPs
- Five I/O channels (one each for disk, tape, display and two for PP communication)
- One tape controller and unit
- One disk controller and unit
- One system console and controller

All pertinent hardware must be operational for TDX to complete a requested task. Either the tape drive or the disk pack containing the library may be used as a deadstart device.

1. CYBER 962/992 systems use the Install-Update CIP utility (IUC). Refer to the appropriate CIP reference manual for information on IUC.

Software Requirements

Disk subsystem controlware must be loaded and executing. If MTS is being used, MTS controlware must be executing. The library device must contain the TDX utility, and a tape containing MSL is required to build the disk library.

TDX expects a deadstart program to be already installed on the deadstart sector of the disk before MSL is installed to the disk. TDX does not install a deadstart program to the deadstart sector, but does modify the sector pointers. Installation of CTI to the disk is one way to install a deadstart program to the deadstart sector.

Restrictions

All command buffers to be copied from tape must contain either a (blank) dollar sign ($) 5553_8 or a double dollar sign ($$) 5353_8 as the first 12 bits of the tape record.

Command buffers must be in display code (6 bit) when written on tape.

Command buffers must not contain a prefix or load table.

Loading Procedures

TDX can be loaded by CMSE after calling MSL from the initial CTI display. When the initial MSL display appears, type TDX and press **(CR)**, or type B, then M, then T from the Manual Operations display.

This loads the disk driver (TDX) into PP1 and the tape driver (TDO) into PP0. The tape driver then loads a third PP with a message driver and most of the messages used by TDX and TDO.

Parameter Entries

All parameter entries are terminated by a **(CR)**. The backspace key (BKSP) allows changing the previous entry.

The following is the initial display presented by TDX.

```
                TDX
DISK AND TAPE TRANSFER UTILITY
        CR TO CONTINUE
```

Press a **(CR)**. The following lines are added to the display.

```
ENTER PARAMETERS

DISK CHANNEL 01
```

Enter the proper disk channel, if different from the default value, and follow it with a **(CR)**. Disk channel may be in range of 0-13 or 20-33. The following line is added to the display.

```
DISK UNIT 00
```

Enter the proper disk unit number (0-77), if different from the default value, and follow it with a **(CR)**. The display clears and the following lines appear.

```
TAPE TYPE 00
0=60X, 1=65X, 2=66X, 3=63X/67X/698
```

Enter the proper tape type (0-3), if different from the default value, and follow it with a **(CR)**. The following line is added to the display.

```
TAPE CHANNEL 13
```

Enter the proper tape channel (0-13, 20-33), if different from the default value, and follow it with a **(CR)**. The following line is added to the display.

```
TAPE EQUIPMENT 00
```

Enter the proper tape equipment number, if different from the default value, and follow it with a **(CR)**. Tape equipment number may be in the range of 0-7 for 63X, 66X, and 67X tape drives and 4-7 for 60X and 65X tape drives. The following line is added to the display.

```
TAPE UNIT 00
```

Enter the proper tape unit number (0-17), if different from the default value, and follow it with a **(CR)**. If the selection entered previously for tape type was 1 (65X), the following line is added to the display. (Otherwise, parameter entry is complete and TDX proceeds to the options display.)

CAUTION

Selecting the same unit number as the deadstart device unit number through a dual access controller using a different channel number is an illegal operation. This causes TDX to hang.

```
TAPE MODE (0=7 TRK, 1=9 TRK) 00
```

Enter the proper tape mode (0 or 1), if different from the default value, and follow it with a **(CR)**. TDX proceeds to the options display. If an illegal parameter is entered for disk channel, tape channel, tape equipment, or tape unit (parameter does not fall within the specified ranges), TDX displays:

```
ILLEGAL ENTRY
```

Pressing the **Space Bar** returns TDX to the parameter display and allows re-entry of the parameter.

If you enter an illegal parameter for tape type and tape mode (parameter is outside the range displayed on the screen), TDX halts and does not continue until you enter a valid command and press **(CR)**. After disk and tape parameters are entered, TDX attempts to connect to the tape drive. If connection to the unit is unsuccessful, TDX displays:

```
TAPE UNIT NOT CONNECTED
```

Pressing the **Space Bar** causes TDX to reattempt to connect to the tape drive.

TDX Options Display

Figure 6-1 is the INITIAL OPTIONS display presented by TDX.

```
A - BUILD MSL ON DISK FROM TAPE
B - BUILD COMMAND BUFFER LIBRARY ON DISK
C - ADD PROGRAMS TO DISK
D - ADD COMMAND BUFFERS TO DISK
E - COPY PROGRAMS TO TAPE
F - COPY COMMAND BUFFERS TO TAPE
G - DISPLAY MSL SYSTEM TABLES
```

Figure 6-1. TDX Initial Options Display

Select the desired option and enter the corresponding letter [no **(CR)** is necessary]. TDX begins execution of the selected option.

TDX Options

The following paragraphs describe the options available on the TDX options display.

A - Build MSL on Disk from Tape

This option allows TDX to create an MSL area on disk and to transfer programs from an MSL tape to disk. TDX must first determine the type of MSL build to be performed. Figure 6-2 is the MSL INSTALLATION OPTIONS display presented by option A.

```
MSL INSTALLATION OPTION

F - FULL MSL/SHARING WITH OS
S - SHORT MSL/SHARING WITH OS
        (810, 815, 825, 830 ONLY)
M - MSL MAINTENANCE ONLY-
        NO OS INFORMATION ON DISK
```

Figure 6-2. MSL Installation Options Display

Selection of option F allows TDX to initialize 20 megabytes of the deadstart disk and install all of MSL.

Selection of option S, for models 810, 815, 825, and 830 only, allows TDX to initialize 12 megabytes of the deadstart disk and install a predefined set of the most frequently used diagnostics.

Selection of option M allows TDX to initialize for program installation to a disk reserved for maintenance programs. Programs are installed to an operator-defined area of the disk. The following line is displayed for option M.

```
STARTING CYLINDER NUMBER 0000
```

You may choose and enter the desired starting cylinder number for the installation (in octal). Enter the desired value, if different from the default value, and follow it with a **(CR)**. If the selected cylinder is large enough to hold the MSL installation, the following message is displayed.

```
SAVE COMMAND BUFFER AREA
Y=YES  N=NO
```

Type Y or N [no **(CR)** is necessary]. If you type Y, TDX initializes the program name table (PNT) and the sector reservation table (SRT) without destroying any command buffer information. If you type N, TDX initializes PNT and SRT, and any command buffers previously written on the selected cylinder are lost. This feature allows the operator to install a new MSL over an old MSL without the necessity of reinstalling the command buffers also.

If the selected cylinder is not large enough to hold the MSL installation, the following message is displayed.

```
NOT ENOUGH ROOM FOR MSL
INSTALLATION. DEADSTART AND
SELECT A NEW CYLINDER NUMBER
```

As indicated, a deadstart is required to escape from this display.

On 895 disk subsystems, the specified MSL area is formatted in small records (322 words). For each of the three options, TDX checks the tracks on which the two tables (PNT and SRT) are to reside and verifies their usability. TDX deletes from the SRT any flawed sectors that are logged in the disk utility map and that lie within the available MSL area.

NOTE

An SRT and PNT are not built for command buffers if option A is selected and if the operator has specified that command buffers should be saved.

TDX allows the operator to specify the first and last programs to be transferred from tape. After creation of the program PNT and SRT, TDX presents the following message.

```
COPY FROM
-CR- = 1ST NAME
```

You may specify the first program to be copied to disk by entering the name of the program and following it with a **(CR)**. As the characters are entered, they appear on the display to the right of COPY FROM. Names are limited to seven characters. Any attempt to enter more than seven characters results in the additional characters successively replacing the seventh character in the name. Pressing only **(CR)** causes TDX to begin copying with the first program it encounters.

After a response has been entered for the CTI message, TDX presents the following message.

```
COPY THRU
-CR- = LAST NAME
```

You may press **(CR)**, which instructs TDX to copy to the last program on the tape, or you may enter the name of the final program to be copied (seven characters or less) and follow it with a **(CR)**.

TDX copies programs from the tape to the disk as directed by the user. The programs copied include CTI, if CTI is on the tape being used.

For tape to disk copies on 844 and 885 disks, TDX has the ability to verify data written to disk. TDX presents the following message.

```
DATA VERIFY (Y/N)
```

Type Y or N [no **(CR)** is necessary]. If you type Y, TDX transfers each block of data from the tape to the disk controller twice. The first transfer is accompanied by a write, and the data is written to the disk. The second transfer is accompanied by a write verify, and the data is checked bit for bit with the data written on the disk.

If you type N data is transferred from tape to disk without any write verification.

NOTE

Use of the write verification feature doubles the time required to perform an MSL program or command buffer library build.

Following the response to the DATA VERIFY message, TDX executes the operation. For each program transferred, the program name is entered into the PNT and the reserved sectors are entered into the SRT. TDX skips over any command buffers located on the tape.

The name of each program copied to disk is displayed while the copy is in progress. The names of programs not copied to disk are not displayed.

Upon completion of the copy operation, TDX displays the number of cylinders used for the build, and the number of cylinders still available. Pressing the **Space Bar** clears the message and allows TDX to display a reduced set of copy options that may be performed without a deadstart.

B - Build Command Buffer Library on Disk

This option allows TDX to create a command buffer area on an MSL disk. Following the selection of this option, TDX reads the starting disk cylinder for MSL from the deadstart sector. TDX creates a command buffer SRT and deletes from it flawed sectors that are logged in the disk utility map and that are within range of the command buffer area. TDX also creates a command buffer PNT.

TDX allows the operator to specify the first and last command buffers to be transferred from tape. After creating the command buffer SRT and PNT, TDX presents the following message.

```
COPY FROM
-CR- = 1ST NAME
```

You may specify the first command buffer to be copied to disk by entering the name of the command buffer and following it with a **(CR)**. As the characters are entered, they appear on the display to the right of COPY FROM. Names are limited to seven characters. Any attempt to enter more than seven characters results in the additional characters successively replacing the seventh character in the name. Pressing only **(CR)** causes TDX to begin copying with the first program it encounters.

After a response has been entered for the COPY FROM message, TDX presents the following message.

```
COPY THRU
-CR- = LAST NAME
```

You may either press **(CR)**, which instructs TDX to copy to the last command buffer on the tape, or enter the name of the final command buffer to be copied (seven characters or less) and follow it with a **(CR)**.

For tape to disk copies on 844 and 885 disks, TDX has the ability to verify data written to disk. TDX presents the following message.

```
DATA VERIFY (Y/N)
```

Type Y or N [no **(CR)** is necessary]. If you type Y, TDX transfers each block of data from the tape to the disk controller twice. The first transfer is accompanied by a write, and the data is written to the disk. The second transfer is accompanied by a write verify, and the data is checked bit for bit with the data written on the disk.

If you type N data is transferred from tape to disk without any write verification.

NOTE

Use of the write verification feature doubles the time required to perform and MSL program or command buffer library build.

Following the response to the DATA VERIFY message, TDX executes the operation. For each command buffer transferred, the name is entered into the command buffer PNT and the reserved sectors are entered into the command buffer SRT. TDX skips over any programs located on the tape.

The name of each command buffer copied to disk is displayed while the copy is in progress. The names of command buffers not copied to disk are not displayed.

Upon completion of the build, TDX displays the number of sectors used for the build and the number of sectors still available. Pressing the **Space Bar** clears the message and allows TDX to display a reduced set of copy options that may be performed without a deadstart.

C - Add Programs to Disk

This option allows TDX to add programs to an MSL area already existing on disk. Following the selection of this option, TDX reads the starting disk cylinder for MSL from the deadstart sector and prepares to receive parameters.

TDX has the ability to check the MSL disk for duplicate names. TDX presents the following message.

```
DUPLICATE NAME CHECK (Y/N)
```

Type Y or N [no **(CR)** is necessary]. If you type Y, TDX presents the following message.

```
REPLACE DUPLICATE NAMES (Y/N)
```

Type Y or N [no **(CR)** is necessary]. If you type Y, then for each program transferred from tape, TDX searches the disk for an existing program with the same name, deletes the disk program if it exists, and copies the new program to disk.

NOTE

Programs deleted or replaced on disk create empty slots in the PNT. These slots are filled by TDX when TDX runs out of room for names at the end of the PNT. Program names added to empty slots appear in the PNT interspersed throughout the program name display.

If you type N in response to REPLACE DUPLICATE NAMES, TDX adds to the disk only those programs that do not already exist there.

If you type N in response to DUPLICATE NAME CHECK, TDX adds all the specified programs to disk regardless of whether or not they already exist there.

TDX allows the operator to specify the first and last programs to be transferred from tape. After selection of the desired duplicate name options, TDX presents the following message.

```
COPY FROM
-CR- = 1ST NAME
```

You may specify the first program to be copied to disk by entering the name of the program and following it with a **(CR)**. As the characters are entered, they appear on the display to the right of COPY FROM. Names are limited to seven characters. Any attempt to enter more than seven characters results in the additional characters successively replacing the seventh character in the name. Pressing only **(CR)** causes TDX to begin copying with the first program it encounters.

After a response has been entered for the COPY FROM message, TDX presents the following message.

```
COPY THRU
-CR- = LAST NAME
```

You may either press **(CR)**, which instructs TDX to copy to the last program on the tape, or enter the name of the final program to be copied (seven characters or less) and follow it with a **(CR)**.

Following the response to the COPY THRU message, TDX executes the operation as specified by the duplicate name parameter selections. For each program transferred, the name of the program is added to the program PNT and the reserved sectors are entered into the program SRT. Names of replaced programs are deleted from the program PNT.

The name of each program added to disk is displayed while the copy is in progress. The names of duplicate programs that are skipped over are not displayed. Following the completion of the operation, TDX displays a reduced set of copy options that may be performed without a deadstart.

D - Add Command Buffers to Disk

This option allows the operator to add command buffers to an MSL area already existing on disk. Following the selection of this option, TDX reads the starting disk cylinder from the deadstart sector and prepares to receive parameters.

TDX has the ability to check the MSL disk for duplicate names. TDX presents the following message.

```
DUPLICATE NAME CHECK (Y/N)
```

Type Y or N [no **(CR)** is necessary]. If you type Y, TDX presents the following message.

```
REPLACE DUPLICATE NAMES (Y/N)
```

Type Y or H [no **(CR)** is necessary]. If you type Y, then for each command buffer transferred from tape, TDX searches the disk for an existing command buffer with the same name, deletes the disk command buffer if it exists, and copies the new command buffer to disk.

NOTE

Command buffers deleted or replaced on disk create empty slots in the command buffer PNT. These slots are filled by TDX when TDX runs out of room for names at the end of the command buffer PNT. Command buffer names added to empty slots show up on the command buffer PNT interspersed throughout the other command buffer names.

If you type N in response to REPLACE DUPLICATE NAMES, TDX adds to the disk only command buffers that do not already exist there.

If you type N in response to DUPLICATE NAME CHECK, TDX adds all specifed command buffers to the disk regardless of whether they already exist there.

TDX allows the operator to specify the first and last command buffers to be transferred from tape. After selection of the desired duplicate name options, TDX presents the following message.

```
COPY FROM
-CR- = 1ST NAME
```

You may specify the first command buffer to be copied to disk by entering the name of the command buffer and following it with a **(CR)**. As the characters are entered, they appear on the display to the right of COPY FROM. Names are limited to seven characters. Any attempt to enter more than seven characters results in the additional characters successively replacing the seventh character in the name. Pressing only **(CR)** causes TDX to begin copying with the first command buffer it encounters.

After a response has been entered for the COPY FROM message, TDX presents the following message.

```
COPY THRU
-CR- = LAST NAME
```

You may either **(CR)**, which instructs TDX to copy to the last command buffer on the tape, or enter the name of the final command buffer to be copied (seven characters or less) and follow it with a **(CR)**.

Following the response to the COPY THRU message, TDX executes the operation as specified by the duplicate name parameter selections. For each command buffer transferred, the name is added to the command buffer PNT and reserved sectors are entered into the command buffer SRT. Names of replaced command buffers are deleted from the command buffer PNT.

The name of each command buffer added to disk is displayed while the copy is in progress. The names of duplicate command buffers that are skipped over are not displayed. Following completion of the operation, TDX displays a reduced set of copy options that may be performed without a deadstart.

E - Copy Programs to Tape

This option allows TDX to copy programs from an MSL disk to tape. Following selection of this option, TDX reads the starting cylinder for MSL from the disk deadstart sector and prepares to receive parameters.

TDX allows the operator to specify the first and last programs to be transferred to tape. Following initialization of the option, TDX presents the following message.

```
COPY FROM
-CR- = 1ST NAME
```

You may specify the first program to be copied to tape by entering the name of the program and following it with a **(CR)**. As the characters are entered, they appear on the display to the right of COPY FROM. Names are limited to seven characters. Any attempt to enter more than seven characters results in the additional characters successively replacing the seventh character in the name. Pressing only **(CR)** causes TDX to begin copying with the first program listed in the program PNT.

NOTE

For disk-to-tape operations, disk programs are read in the order in which they appear in the program PNT.

After a response is entered for the COPY FROM message, TDX presents the following message.

```
COPY THRU
-CR- = LAST NAME
```

You may either press **(CR)**, which instructs TDX to copy to the last name in the PNT, or enter the name of the final program to be copied (seven characters or less) and follow it with a **(CR)**.

TDX allows you to select the desired density for the disk-to-tape copy on all 66X/67X tape drives. If, however, you are using a 679 tape drive, you must first select which type of 679 drive is being used for the copy. TDX presents the following message.

```
679 TAPE OPTIONS
0 = 679-2,3,4, and 63X
1 = 679-5,6,7, and 698
```

NOTE

The 679-5,6,7 tape drives have group coded recording (GCR) capabilities.

TDX then presents one of the following messages, depending upon which type of 66X or 67X tape drive is being used.

```
667  TAPE DENSITY OPTIONS
0 = 556 NRZI
1 = 800 NRZI

669  TAPE DENSITY OPTIONS
0 = 800 NRZI
1 = 1600 PE

677  TAPE DENSITY OPTIONS
0 = 556 NRZI
1 = 800 NRZI

679  TAPE DENSITY OPTIONS
0 = 800 NRZI
1 = 1600 PE

679  TAPE DENSITY OPTIONS
0 = 1600 PE
1 = 6250 GCR
```

NOTE

If a 679-5,6,7 tape drive is cabled to a 67X controller that doesn't have GCR capabilities and you selected 6250 GCR recording mode, the following error message appears.

```
NO GCR CAPABILITY
```

If this message appears, reselect 1600 PE density.

TDX allows the operator either to create a new tape of MSL programs or add programs to an existing tape. TDX presents the following message.

```
IS DATA ON THE TAPE VALID (Y/N)
```

Type `Y` or `N` [no (CR) is necessary]. If you type `Y`, TDX searches the tape for a tapemark or a filemark which signals the end of valid data on tape. TDX backspaces over the filemark and prepares to write the first block of new data.

If you type `N`, TDX returns to the beginning of the tape and prepares to write the first block of new data there. Data previously written on the tape is destroyed.

Following a response to IS DATA ON THE TAPE VALID message, TDX executes the operation. The name of each program copied to tape is displayed while the copy is in progress. At the completion of the operation, TDX displays a reduced set of copy options that may be performed without a deadstart.

F - Copy Command Buffers to Tape

This option allows TDX to copy command buffers from an MSL disk to tape. Following the selection of this option, TDX reads the starting cylinder for MSL from the disk deadstart sector and prepares to receive parameters.

TDX allows the operator to specify the first and last command buffers to be transferred to tape. Following initialization of this option, TDX presents the following message.

```
COPY FROM
-CR- = 1ST NAME
```

You may specify the first command buffer to be copied to tape by entering the name of the command buffer and following it with a **(CR)**. As the characters are entered, they appear on the display to the right of COPY FROM. Names are limited to seven characters. Any attempt to enter more than seven characters results in the additional characters successively replacing the seventh character in the name. Pressing only **(CR)** causes TDX to begin copying with the first command buffer listed in the command buffer PNT.

NOTE

For disk-to-tape operations, disk command buffers are read in the order in which they appear in the disk command buffer PNT.

After a response has been entered for the COPY FROM message, TDX presents the following message.

```
COPY THRU
-CR- = LAST NAME
```

You may either press **(CR)**, which instructs TDX to copy to the last name in the PNT, or enter the name of the final program to be copied (seven characters or less) and follow it with a **(CR)**.

TDX allows you to select the desired density for the disk-to-tape copy on all 66X/67X tape drives. If, however, you are using a 679 tape drive, you must first select which type of 679 drive is being used for the copy. TDX presents the following message.

```
679 TAPE OPTIONS
0 = 679-2,3,4
1 = 679-5,6,7
```

NOTE

The 679-5,6,7 tape drives have group coded recording (GCR) capabilities.

TDX then presents one of the following messages, depending upon which type of 66X or 67X tape drive is being used.

```
667  TAPE DENSITY OPTIONS
0 = 556 NRZI
1 = 800 NRZI

669  TAPE DENSITY OPTIONS
0 = 800 NRZI
1 = 1600 PE

677  TAPE DENSITY OPTIONS
0 = 556 NRZI
1 = 800 NRZI

679  TAPE DENSITY OPTIONS
0 = 800 NRZI
1 = 1600 PE

679  TAPE DENSITY OPTIONS
0 = 1600 PE
1 = 6250 GCR
```

NOTE

If a 679-5,6,7 tape drive is cabled to a 67X controller that doesn't have GCR capabilities and you selected 6250 GCR recording mode, the following error message appears.

```
NO GCR CAPABILITY
```

If this message appears, reselect 1600 PE density.

TDX allows the operator to either create a new tape of command buffers or add command buffers to an existing tape. TDX presents the following message.

```
IS DATA ON THE TAPE VALID (Y/N)
```

Type Y or N [no **(CR)** is necessary]. If you type Y, TDX searches the tape for a tapemark or filemark which signals the end of valid data on tape. TDX backspaces over the filemark and prepares to write the first block of new data.

If you type N, TDX returns to the beginning of the tape and prepares to write the first block of new data there. Data previously written on the tape is destroyed.

Following a response to IS DATA ON THE TAPE VALID message, TDX executes the operation. The name of each command buffer copied to tape is displayed while the copy is in progress. At the completion of the operation, TDX displays a reduced set of copy options that may be performed without a deadstart.

G - Display MSL System Tables

This option causes TDX to display the PNTs and SRTs for both programs and command buffers. Successive **(CR)**s are required to page through the SRT and PNT.

Operator Entries

During execution, TDX recognizes certain special keyboard entries and special restrictions for making those keyboard entries. The special restrictions are:

- If all possible responses to a TDX display are listed on that display (that is, Y/N), only those characters are acceptable. TDX ignores all other character entries.
- When TDX requests input of a numerical value, an octal value is expected and only octal digits are accepted.
- A **(CR)** is necessary to end the entry of all parameters that allow more than one character input. A **(CR)** must follow disk and tape parameter entries. Those displays that require only a single character entry execute immediately upon receiving a valid character.

Following are special characters recognized by TDX.

Character	Description
Backspace	**Backspace** moves the cursor one space to the left and may be used to re-enter a character for those entries that end with a (CR). **Backspace** is not recognized by entries requiring a single character.
Space Bar	**Space Bar** may be used to resume TDX execution following a halt for error display, information display, or user-requested halt.
S	The S key may be used to halt TDX during the transfer of programs or command buffers. TDX halts following the transfer of the current program. If TDX is resumed, it resumes at the point at which it was halted.
R	The R key may be used to restart TDX during the transfer of programs or command buffers. TDX finishes the current transfer, updates the PNT and SRT, and proceeds to the appropriate end-of-copy display.
G	The G key performs the G option (DISPLAY SYSTEM TABLES) presenting the PNT/SRT display. The G key is only valid during tape parameter entry or when it is listed as part of an options display. The G key allows the operator to debug disk problems before connecting to a valid tape drive.

Informative Messages

Following are messages presented by TDX during normal execution.

Message	Description
FORMATTING	This message applies to the 895 disk subsystem. It indicates that TDX is formatting the MSL area in small records (322 words per record).
INITIALIZING	Indicates that TDX is checking the starting cylinder for flaws, or is creating an SRT and PNT prior to installing programs or command buffers on the disk.
CYLINDERS USED THIS BUILD =	Indicates the octal value of the number of cylinders used to build the MSL program area.
CYLINDERS STILL AVAILABLE =	Indicates the octal value of the number of cylinders still available in the MSL program area.
SECTORS USED THIS BUILD =	Indicates the octal value of the number of sectors used to build the command buffer area.
SECTORS STILL AVAILABLE =	Indicates the octal value of the number of sectors still available in the command buffer area.
HALTED xxxxxxx	This message indicates that an S key has been entered which has halted execution of TDX. The xxxxxxx in this message is the name of the last program or command buffer copied by TDX. Pressing the **Space Bar** clears this message and causes TDX to resume execution at the point where it was halted.
INSTALLATION COMPLETE DEADSTART IS REQUIRED	Indicates TDX has completed a disk build for automatic CIP tape installation. Deadstart to continue.

Error Messages

Following are the error messages presented by TDX.

Message	Description
ILLEGAL BUILD SELECTION OS FILES COULD BE DESTROYED	Indicates the build option selected could cause operating system files to be destroyed because space previously allocated to the operating system is being used. Choose an installation mode that will not destroy operating system files or deadstart and release the disk space using CTI.
ILLEGAL ENTRY	The user entered an illegal parameter during parameter entry. Press the **Space Bar** to return to the parameter display and re-enter the parameter.
UNUSABLE DISK	Indicates that the default starting cylinder for a HIVS installation is faulty. The operator must deadstart and perform the installation to a different device.
MSL STARTING CYLINDER UNUSABLE	Indicates that the starting cylinder and the two succeeding cylinders are unsuitable for a maintenance-only installation. A deadstart is required to reattempt the installation at another cylinder.
MSL STARTING CYLINDER UNUSABLE ENTER -CR- TO USE ALTERNATE CYLINDER yyyy OR RELOAD TDX AND SELECT A NEW CYLINDER	Indicates that the starting cylinder is unusable, although one of the two succeeding cylinders is suitable for the operation. The yyyy is the cylinder which TDX has found to be suitable. Entering a **(CR)** allows TDX to prepare cylinder yyyy for the operation. Entering any other character allows TDX to request another starting cylinder.
IMPOSSIBLE TO INSTALL PROGRAMS AND SAVE COMMAND BUFFER AREA	The operator is saving a command buffer library at cylinder xxxxxxx. TDX does not examine the two succeeding cylinders to find a suitable starting cylinder. A deadstart is required.
xxxxxxx NOT FOUND	Indicates that TDX has not been able to locate a program or command buffer for which it has been searching. The xxxxxxx in this message is the name being searched for. In the case of a tape-to-disk copy, the TDX search is initiated by a COPY FROM request. In the case of a disk-to-tape copy, the TDX search may be initiated by either a COPY FROM or COPY THRU request. For a COPY THRU request, TDX begins the search with the program entered for the COPY FROM message. Pressing the **Space Bar** returns TDX to the copy message that contains the unknown name.

Message	Description
xxxxxxx NO TAPE WRITE RING	Indicates that a disk-to-tape copy is being attempted and that no write ring is being detected on the tape. You may, upon seeing the NO TAPE WRITE RING message, unload and dismount the tape, insert a write ring into the tape hub, and mount and reload the tape. When you press the **Space Bar**, the tape is positioned at the beginning of tape and the copy proceeds. The xxxxxxx is the name of the program or command buffer that TDX was working with.
xxxxxxx TAPE FUNC REJ yyyy	Indicates that a function sent to the tape drive or data channel converter (60X or 65X) has not been accepted. The xxxxxxx in this message is the name of the program or command buffer that TDX was working with when the error occurred, and the yyyy is the octal value of the function code that was rejected. TDX tries the operation three times (including timeouts) before displaying the message. Pressing the **Space Bar** allows TDX to retry the operation.
xxxxxxx DCC ERR STAT yyyy	Indicates that the status received from the data channel converter for a tape drive (60X or 65X) shows that an error condition exists. The xxxxxxx in this message is the name of the program or command buffer that TDX was working with when the error occurred, and yyyy is the octal status word. Pressing the **Space Bar** allows TDX to request the current status word.
xxxxxxx TAPE ERR STAT yyyy	Indicates that the status received from the tape drive shows that an error condition exists. The xxxxxxx in this message is the name of the program or command buffer that TDX was working with when the error occurred, and yyyy is the octal status word. Pressing the **Space Bar** allows TDX to attempt to continue the operation, though the result may not be reliable.
xxxxxxx TAPE UNIT NOT READY	Indicates that the status received from the tape drive shows that the unit is not ready. The xxxxxxx is the name of the program or command buffer that TDX was working with. Correct the not ready condition without moving the tape and press the **Space Bar** to continue.
xxxxxxx NAME TOO LONG	Indicates that TDX has detected a program or command buffer name on tape that contains more than seven characters. The xxxxxxx in the message is the first seven characters of the name that is too long. Pressing **Space Bar** allows TDX to skip to the next program or command buffer and continue the operation.

Message	Description
xxxxxxx COMMAND TOO LONG	Indicates that during a tape-to-disk copy, TDX has encountered a command to be placed on disk that is greater than 60 (decimal) characters long. The xxxxxxx in the message is the name of the program or command buffer where the faulty command was found. Pressing the **Space Bar** allows TDX to truncate the command to 60 (decimal) characters and continue the operation.
xxxxxxx NOT COPIED - END OF TAPE	Indicates TDX encountered the end of tape while writing program xxxxxxx in a disk to tape copy. TDX backspaced the tape and wrote end of information and file markes to the tape before displaying the message. A deadstart is required.
DISK BUSY	Indicates that the disk general status has responded busy to 10000 (octal) attempts by TDX to perform a seek to read or write. Pressing the **Space Bar** allows TDX to continue the read or write attempt.
DISK CONTROLLER RESERVED	Indicates that the disk controller general status shows the multiple access disk controller continues to be reserved to another PP channel following 20 (decimal) attempts to connect to the unit. TDX continues to display the message and attempt the connect until successful or until a deadstart is performed.
DISK UNIT RESERVED	Indicates that the disk general status shows that the disk remains reserved to another controller following 20 (decimal) attempts to connect to the disk. TDX continues to display the message and attempt the connect until successful or until a deadstart is performed.
DISK FUNC REJ yyyy xxxxxxx	Indicates that a function sent to the disk controller has been rejected. The xxxxxxx in this message is the name of the program or command buffer that TDX was working with when the error occurred, and the yyyy is the octal value of the function code that was rejected. Pressing the **Space Bar** allows TDX to retry the operation.
DISK ERR STAT yyyy xxxxxxx	Indicates that the status received from the disk drive shows that an error condition exists. The xxxxxxx in the message is the name of the program or command buffer that TDX was working with when the error occurred, and yyyy is the octal status word. Pressing the **Space Bar** allows TDX to continue the operation, though the result may not be reliable.
PNT FULL xxxxxxx	Indicates that the disk PNT is full. The xxxxxxx in the message is the name of the program or command buffer that filled the PNT. A deadstart is required to clear this message.

Message	Description
SRT FULL xxxxxxx	Indicates that the disk SRT has reserved the entire available area on the disk. The xxxxxxx in this message is the name of the program or command buffer that filled the disk. A deadstart is required to clear this message.
FLAW CYL xxxx TRK yyyy SEC zzzz	Indicates that TDX has failed in four consecutive attempts to write data to a disk sector. The values in the message are the cylinder (xxxx), track (yyyy), and the sector (zzzz), which may not be written in. Pressing the **Space Bar** allows TDX to continue the copy with another sector. The bit of the flawed sector remains set in the SRT to ensure that TDX will not attempt to use the bad sector again.
COPY ERROR xxxxxxx	Indicates that during a copy operation, the program or command buffer xxxxxxx could not be copied successfully. Pressing the **Space Bar** allows TDX to skip to the next program or command buffer and resume copying.
DEADSTART SECTOR ERROR	Indicates TDX was unable to read or write the deadstart sector. A deadstart is required.
DISK CONTROLLER TRANSFER ERROR xxxxxxx	Indicates TDX was unable to output or input the expected number of words to or from the disk controller, but that the general status indicates no errors. The xxxxxxx is the name of the program or command buffer being copied. Pressing the **Space Bar** causes TDX to retry the transfer.
INTER-PP DATA TRANSFER ERROR	Indicates the tape or disk driver is unable to output or input the expected number of words to or from the other driver. The xxxxxxx is the name of the current program or command buffer. A deadstart is required.
FORMATTING ERROR	An error occurred while formatting the MSL area on an 895 disk. A deadstart is required.

Hardware Initialization and Verification Software Tape-to-Disk Utility (HIVS TDX)

The hardware initialization and verification software (HIVS) version of TDX allows the operator to build a HIVS library on disk.

Hardware Requirements

The HIVS version of TDX requires the following hardware for proper execution.

- Three PPs
- Five I/O channels (one each for disk, tape, and display, and two for PP communication)
- One tape controller and unit
- One disk controller and unit
- One system console and controller

All pertinent hardware must be operational for the HIVS version of TDX to complete a requested task. Either the tape drive or the disk pack containing the library may be used as a deadstart device.

Software Requirements

Disk subsystem controlware must be loaded and executing. If MTS is being used, MTS controlware must be executing. The library device must contain the HIVS version of the TDX utility, and a tape containing HIVS is required to build the disk library.

TDX expects a deadstart program to be already installed on the deadstart sector of the disk before MSL is installed to the disk. TDX does not install a deadstart program to the deadstart sector, but does modify the sector pointers. Installation of CTI to the disk is one way to install a deadstart program to the deadstart sector.

Loading Procedures

HIVS TDX is called into operation by choosing the T option on the CTI MANUAL OPERATIONS display, or by selecting M, then TDX and pressing **(CR)**.

Parameter Entries

Following is the initial display presented by the HIVS version of TDX.

```
ENTER PARAMETERS

DISK CHANNEL 01
```

Enter the proper disk channel, if different from the default value, and follow it with a **(CR)**. The following line is added to the display.

```
DISK UNIT 00
```

Enter the proper disk unit number, if different from the default value, and follow it with a **(CR)**. The display clears and the following lines appear.

```
TAPE TYPE 00
0=60X, 1=65X, 2=66X, 3=63X/67X
```

Enter the proper tape type, if different from the default value, and follow it with a **(CR)**. The following line is added to the display.

```
TAPE CHANNEL 13
```

Enter the proper tape channel, if different from the default value, and follow it with a **(CR)**. The following line is added to the display.

```
TAPE EQUIPMENT 00
```

Enter the proper tape equipment number, if different from the default value, and follow it with a **(CR)**. The following line is added to the display.

```
TAPE UNIT 00
```

Enter the proper tape unit number, if different from the default value, and follow it with a **(CR)**. If the selection entered previously for tape type was 1 (65X), the following line is added to the display (otherwise, parameter entry is complete and the HIVS version of TDX proceeds to the options display).

```
TAPE MODE (0=7 TRK, 1=9 TRK)
```

Enter the proper tape mode, if different from the default value, and follow it with a **(CR)**. The HIVS version of TDX presents the following message as it prepares the disk for the HIVS transfer.

```
INITIALIZING
```

Immediately following completion of the initialization process, the HIVS version of TDX begins the transfer of HIVS programs from tape to disk. The following message is presented while the copy is in process.

```
LOADING
```

If CTI exists on the tape with the HIVS library, the HIVS version of TDX automatically skips over CTI and copies only the HIVS programs. When the copy is complete, the HIVS version of TDX displays the following message.

```
INSTALLATION COMPLETE
DEADSTART IS REQUIRED
```

Special Keyboard Entries

During execution, the HIVS version of TDX recognizes certain special keyboard entries and special restrictions for making those entries. The special restrictions are:

- When the HIVS version of TDX requests input of a numerical value, an octal value is expected and only octal digits are accepted.
- A **(CR)** must follow disk and tape parameter entries.

Following are special characters recognized by the HIVS version of TDX.

Character	Description
Backspace	**Backspace** moves the cursor one space to the left and may be used to re-enter a character for those entries that end with a **(CR)**.
Space Bar	**Space Bar** may be used to resume the HIVS version of TDX execution following a halt for error display or user requested halt.
S	The S key may be used to halt the HIVS version of TDX during the transfer of HIVS programs. HIVS TDX halts following the transfer of the current program. If HIVS TDX is resumed, it resumes at the point at that it was halted.
R	The R key may be used to restart the HIVS version of TDX during the transfer of programs. HIVS TDX finishes the current transfer and proceeds to the END display.

Error Messages

Following are the error messages presented by the HIVS version of TDX.

Message	Description
`UNUSABLE DISK`	Indicates that the default starting cylinder for a HIVS installation is faulty. You must deadstart and perform the installation to a different device.
`TAPE FUNC REJ yyyy`	Indicates that a function sent to the tape drive or data channel converter (60X or 65X) has not been accepted. The yyyy in the message is the octal value of the function code that was rejected. The HIVS version of TDX tries the operation three times (including timeouts) before displaying the message. Pressing the **Space Bar** allows HIVS TDX to retry the operation.
`DCC ERR STAT yyyy`	Indicates that the status received from the data channel converter for a tape drive (60X or 65X) shows that an error condition exists. The yyyy in the message is the octal status word. Pressing the **Space Bar** allows the HIVS version of TDX to request the current status word.

Message	Description
TAPE ERR STAT yyyy	Indicates that the status received from the tape drive shows that an error condition exists. The yyyy in the message is the octal status word. Pressing the **Space Bar** allows the HIVS version of TDX to attempt to continue the operation, though the result may not be reliable.
DISK BUSY	Indicates that the disk general status has responded busy to 10000 (octal) attempts by the HIVS version of TDX to perform a seek to read or write. Pressing the **Space Bar** allows HIVS TDX to continue the read or write attempt.
DISK CONTROLLER RESERVED	Indicates that the disk controller general status shows the multiple access disk controller is reserved to another PP channel following 20 (decimal) attempts to connect to the unit. The HIVS version of TDX continues to display the message and attempt the connect until successful or until a deadstart is performed.
DISK UNIT RESERVED	Indicates that the disk general status shows that the disk is reserved to another controller following 20 (decimal) attempts to connect to the disk. The HIVS version of TDX continues to display the message and attempt the connect until successful or until a deadstart is performed.
DISK FUNC REJ yyyy	Indicates that a function sent to the disk controller has been rejected. The yyyy in the message is the octal value of the function code that was rejected. Pressing the **Space Bar** allows the HIVS version of TDX to retry the operation.
DISK ERR STAT yyyy	Indicates that the status received from the disk drive shows that an error condition exists. The yyyy in the message is the octal status word. Pressing the **Space Bar** allows the HIVS version of TDX to continue the operation, though the result may not be reliable.
PNT FULL	Indicates that the disk PNT is full. A deadstart is required to clear this message.
SRT FULL	Indicates that the disk SRT has reserved the entire available area on the disk. A deadstart is required to clear this message.
FLAW CYL xxxx TRK yyyy SEC zzzz	Indicates that the HIVS version of TDX has failed in four consecutive attempts to write data to a disk sector. The values in the message are the cylinder (xxxx), track (yyyy), and the sector (zzzz), which may not be written in. Pressing the **Space Bar** allows HIVS TDX to continue the copy with another sector. The bit of the flawed sector remains set in the SRT to ensure that HIVS TDX will not attempt to use the bad sector again.

Capture Buffer Dump Utility (CBD)

CBD is a capture vehicle formatting program for CYBER model 990 and for CYBER 994 only. The 512-word capture buffer is moved to CM, then formatted and printed. CBD also provides a means of loading PFC memory and image memory.

Hardware Requirements

CBD requires the following hardware.

- One PP
- One 512 or 580 line printer
- One library loading device
- A 512-word block of CM
- One model 990 CPU

Software Requirements

CBD loads and runs under CMSE. CBD uses the move capture buffer (MX) command.

Loading Procedure and Parameters

CBD is loaded into a PP using either the LT or CP command, as in the following LT example:

```
LT,CBD,,ccee,cmup,cmlo,auto,ident,sadr,eadr
```

where the following parameters apply:

ccee	Channel and equipment of the line printer.
cmup	Upper half of CM address to use.
cmlo	Lower half of CM address to use.
auto	Allows automatic selection of operation: 1 = Load PFC memory (580 only). 2 = Load image memory. 4 = Dump capture buffer. 10 = Omit page headings after first page. If 1, 2, or 4 is not present, a parameter stop occurs.
ident	A hexadecimal value that is printed as a dump identifier.
sadr	Starting capture buffer address to be printed.
eadr	Ending capture buffer +1 address to be printed (zero prints to entry 512).
eadr	Ending capture buffer +1 address to be printed (zero prints to entry 512).

CAUTION

If the ccee parameter is present, all remaining parameters must be supplied or they are taken as zero values.

The LT command cannot be used to change parameters MCP12 (printer type) or MCP13 (train type). The following load sequence can be used if these parameters require change.

Command	Definition	
DPN		
CPn,CBD		
PPn		
EPn,1002,ccee	1205	
EPn,1003,cmup	737	Upper block of 16 MB but below
EPn,1004,cmlo	7000	CMSE communication area.
EPn,1005,auto	0000	Parameter stop occurs.
EPn,1006,ident	0000	
EPn,1007,sadr	0000	All 512 entries to be
EPn,1010,eadr	0000	printed.
EPn,1012,printer	2	580
EPn,1013,train	6	
RUn,100		

Initial Display

If no auto operation bits are set in MCP5, the display in figure 6-3 appears.

```
CAPTURE BUFFER DUMP UTILITY
(NEEDS 512 WORD BLOCK OF CM)

TYPE C - TO DUMP CAPTURE BUFFER
TYPE I - TO LOAD IMAGE MEMORY
TYPE L - TO LOAD PFC MEMORY
TYPE D - TO END CBD
TYPE R - TO RESTART
1002 = LP(CCEE)
1003 = FWA OF 512 WORD CM (UPPER)
1004 = FWA OF 512 WORD CM (LOWER)
1005 = AUTO CONT. BITS (1=LOAD PFC,
       2=LOAD IMAGE, 4=DUMP CAP BUF,
       10=OMIT PAGE HEADING REPEAT)
1006 = CAPTURE BUFFER IDENT. (HEX)
1007 = CAPTURE BUF START ADRS. (HEX)
1010 = CAPTURE BUF END ADRS. (HEX)
1011 = LINES/INCH (10=8LPI, 11=6LPI)
1012 = PRINTER (1=512, 2=580)
1013 = TRAIN   (1, 2, 3, 4, 5, OR 6)
```

Figure 6-3. Capture Buffer Dump Utility Display

Error Messages

The following message appears while dumping the capture buffer.

```
DUMPING CAPTURE BUFFER
```

The following message appears when the line printer is not ready. Check for a paper-out condition. After making the line printer ready, press the **Space Bar** to continue.

```
LP NOT READY
```

The following message appears during a PFC load when a 6/8 line coincidence cannot be achieved after five single spaces of the line printer.

```
UNABLE TO GET COINCIDENCE - TYPE R
```

The following message appears during an image memory load when an illegal printer type was previously selected. After changing address 1012, pressing the **Space Bar** clears the message.

```
ILLEGAL PRINTER TYPE - RESET 1012
```

The following message occurs during an image memory load when an illegal image type was previously selected. After changing address 1013, pressing the **Space Bar** clears the message.

```
ILLEGAL TRAIN TYPE - RESET 1013
```

Capture Buffer Dump Format

Figure 6-4 is a sample of the printed output.

```
CAPTURE VEHICLE FORMATTED DUMP - IDENTIFIER = 0000

                            ABRTOC        ALLCLR
                            VECVLD        LSUENA
       ISSUED               VECACT        LSUENX
       CIRVLD        CREQUA SEGTRD RESPOA RPLENA        LSULPT
       CWDVLD        CREQOB PAGTRD RESPOB RPLENX RPLVLD LSUVLD ACUVLD        ISSTRD
ADR    CWAVLD CSADDR CREQOC CACHIT RESPOC FCTNGO RPLTAG LSUTAG ACUTAG ENTCTR EPSTRT PRTISS
------------------------------------------------------------------------------------------
000 T-  0000   000    000   000000  000   000000  0-0    00-0   0-0     0      00     00
--- -E  1111   ---    111   111111  111   111111  1--    11--   1--     -      11     --
511 ||         7FF                                  F       F     F     F             FF
    ||
    ||
    ||___(Enable Entry)
    |____(Start Capture)
```

Figure 6-4. Sample Display for SOURCE Directive

Appendixes

Glossary A

A

A register
Address register.

Abs
Absolute.

AC
Address control.

Adrs
Address.

AEP
Advanced emulation package. Provides dual screen support for CC598-A/B consoles.

ALN
Arithmetic and logical network.

ALU
Arithmetic and logical unit.

AOR
Address out of range.

Arith
Arithmetic.

ASCII
American Standard Code for Information Interchange.

ASE
Address specification error.

ASID
Active segment identifier.

Assy
Assembly.

ATS
Advanced tape subsystem.

Aux
Auxiliary.

B

BCD
Binary coded decimal.

BD
Branch delta.

BDP
Business data processor.

Bfr
Buffer.

Bin
Binary.

Bkpt
Breakpoint.

BN
Byte number.

Bound
Boundary.

Br
Branch.

C

CA
Condition action.

CBD
Capture buffer dump utility.

CBU
Command buffer editing utility.

CABR
Cache address buffer register.

CAE
Condition action extension.

CAR
Cache address register.

CBP
Code base pointer.

CCR

Clock condition register.

CDA

Refer to Common Disk Area.

CEL

Critical error log.

Chan

Channel.

CIP

CYBER Initialization Package.

Clk

Clock.

Clr

Clear.

CM

Central memory.

CMC

Central memory control.

CMR

Central memory resident.

CMSE

Common maintenance software executive.

Cnt

Count.

Cntr

Counter.

Coef

Coefficient.

Com

Common.

Common Disk Area (CDA)

The disk storage area that contains a default parameter block, EI, microcode, and CEL.

Comp

Complement.

Cond

Refer to Condition.

Condition (Cond)

A test within a subsection that uses a particular set of operands to test the common hardware element.

Confl

Conflict.

Cont

Control.

Conv

Convert, Converter.

Cor

Corrected.

CPU

Central processing unit.

CS

Condition sensing.

CST

Control store.

CTI

Common test and initialization.

CW1

Control word 1.

CW2

Control word 2.

D

DAI

Data interchange.

DCDR

Decoder.

DCI

Data control information. Two bytes of information that give the maintenance register address.

DEC

Dependent environment control, decimal.

Decr

Decrement.

Descr
Descriptor.

Destn
Destination.

DDP
Distributive data path.

DEX
Diagnostic executive (interface).

Disassy
Disassembly.

Div
Divide.

Dlyd
Delayed.

Dsbl
Disable.

DSC
Display controller.

DSRB
Diagnostic software release bulletin.

E

EBCDIC
Extended binary coded decimal interchange code.

ECC
Error correction code.

EC1
Error code 1.

EC2
Error code 2.

ECL
Emitter-coupled logic.

ECM
Extended central memory.

ECR
Environment control register.

EDS
Extended deadstart.

EI
Environment interface.

EM
Exit mode.

EMMR
Exit mode mask register.

Enbl
Enable.

Encdr
Encoder.

ESE
Environment specification error.

EXC
CPU monitor program.

EXCH
Exchange.

Exp
Exponent.

Ext
External.

FCA
Field change announcement.

Fctn
Function.

F

FF
Flip flop.

FIFO
First-in, first-out.

FL
Field length.

FLC
CYBER 170 mode central memory field length.

FMD

Fixed module drive.

FNO

Fanout.

FP

Floating point.

FRC

Free running counter.

FU

Functional unit.

FWA

First word address.

G

GE

Greater than or equal.

Gen

Generator.

Genl

General.

GT

Greater than.

H

Hldg

Holding.

HDT

Hardware descriptor table.

HIVS

Hardware initialization and verification software.

HVS

Hardware verification sequencer.

I

IC

Integrated circuit.

ICC

Instruction completion control.

ICP

Instruction control pipeline.

IDX

Indexed.

IF

Instruction fetch.

II

Instruction issue.

Immed

Immediate.

Incr

Increment.

Indef

Indefinite.

Inf

Infinite.

Inh

Inhibit.

Inp

Input.

Instr

Instruction.

Int

Internal.

Intrpt

Interrupt.

I/O

Input/output.

IOU

Input/output unit.

ISE

Instruction specification error.

IU

Instruction unit.

J

JPS

Job process state pointer.

K

K/L

Key/lock.

Kypt

Keypoint.

L

Ld

Load.

LM

Local memory.

LOS

Loss of significance.

LRU

Least recently used.

LSD

Least significant digit.

LSI

Large scale integration.

LSM

Last symbol mask.

LT

Less than.

Lwr

Lower.

M

MAC
Maintenance access control.

Maint
Maintenance.

MAR
Micrand address register.

MBE
Multiple bit error.

MC
Master clear.

MCH
Maintenance channel.

MCR
Monitor condition register.

MCU
Maintenance control unit.

Mem
Memory.

Micr
Micrand.

Misc
Miscellaneous.

MMR
Monitor mask register.

MOP
Micro-operation.

MPS
Monitor process state.

MR
Maintenance register.

MSB
Most significant bit.

MSC
Micrand sequence control.

MSD

Most significant digit.

MSL

Maintenance software library.

MSNZB

Most significant nonzero byte.

MSS

Mass storage system.

MTS

Magnetic tape subsystem.

Mult

Multiply.

Mux

Multiplexer.

N

NE

Not equal.

Neg

Negative.

Network

A hardware element wholly contained on a pack and sharing no common circuits with any other network.

Norm

Normalize.

NOS

Network operating system.

ns

Nanosecond.

O

OCR

Optical card reader.

OP

Operand.

Opcode

Operation code.

OPI

Operand issue.

Opn

Operation.

OS

Operating system.

OSB

Operating system bounds register.

Out

Output.

Ovfl

Overflow.

P

P

Program address.

Pack

A replaceable hardware module.

Par

Parity.

PDM

Processor detected malfunction.

PE

Parity error.

PFA

Page frame address.

PFS

Processor fault status.

Ph

Phase.

PIT

Process interval timer.

PMF

Performance monitoring facility.

PN

Page number.

PNT

Program name table.

PO

Page offset.

PONR

Point of no return.

Pos

Positive.

PP

Peripheral processor.

PPM

Peripheral processor memory.

PPU

Peripheral processor unit.

P register

Program address register.

Prev

Previous.

Pri

Priority.

Prod

Product.

PSM

Page size mask.

PSWF

Page table search without find.

PTA

Page table address.

PTE

Page table entry.

PTL

Page table length.

PTM

Processor test mode.

PVA

Process virtual address.

PW

Partial write.

Q

Quad

Quadrant.

R

RA

Reference address.

RA/FL

Reference address/field length.

RAC

CYBER 170 mode central memory reference address.

RAM

Random access memory.

RCL

Read and clear lock.

Rcvr

Receiver.

RDS

Register/data select.

Ref

Reference.

Regen

Regenerator.

Rel

Relative.

Rem

Remainder.

Req

Request.

Resync

Resynchronize, resynchronizer, resynchronization.

RF

Register file.

Rgtr
Register.

RSL
Read and set lock.

R/W
Read/write.

RMA
Real memory address.

RMS
Rotating mass storage.

ROM
Read only memory.

S

SBE
Single bit error.

SC
Soft control.

SCM
Soft control memory.

SCT
Special character table.

SECDED
Single error correction/double error detection.

Seg
Segment.

Sel
Select.

Sep
Separate.

Seq
Sequence.

Sig
Significant.

SIT
System interval timer.

SM

Segment map.

Spec

Specification.

SPID

Segment/page identifier.

SRT

Sector reservation table.

SS

Status summary.

STA

Segment table address.

STL

Segment table length.

Str

Stream.

Subsection

A series of tests that check out a specified hardware element.

Subtr

Subtract.

Suppr

Suppression.

SV

Specification value.

SVA

System virtual address.

Sw

Switch.

Syn

Syndrome.

Sync

Synchronize.

T

TDX
Tape-to-disk utility.

Termn
Terminate.

Test
A general term that can refer to conditions, subsections, sections, or units.

TFA
Tag file address.

U

ubit
Micrand bit.

UCEL1
Uncorrected error log 1.

UCEL2
Uncorrected error log 2.

UCR
User condition register.

UEM
Unified extended memory.

UMR
User mask register.

Unbr
Unbranch.

Uncond
Unconditional.

Uncor
Uncorrectable.

Undfl
Underflow.

Unit
An arbitrary functional area within the processor.

Unlog
Unlogged.

Upr
Upper.

usec
Microsecond.

UTP
Untranslatable pointer.

V

VLEX
Virtual level executive interface.

VMID
Virtual machine identifier.

W

WDS
Write data select.

WOI
Word of interest.

X

Xfer
Transfer.

Xltr
Translator.

Xmtr
Transmitter.

Z

ZIF
Zero insertion force.

ZF
Zero flag.

Character Set B

The characters that may be used with CMSE are indicated in the following character set.

CDC Graphic	ASCII Graphic	Display Code	026 Punch	ASCII Code
A	A	01	12-1	41
B	B	02	12-2	42
C	C	03	12-3	43
D	D	04	12-4	44
E	E	05	12-5	45
F	F	06	12-6	46
G	G	07	12-7	47
H	H	10	12-8	48
I	I	11	12-9	49
J	J	12	11-1	4A
K	K	13	11-2	4B
L	L	14	11-3	4C
M	M	15	11-4	4D
N	N	16	11-5	4E
O	O	17	11-6	4F
P	P	20	11-7	50
Q	Q	21	11-8	51
R	R	22	11-9	52
S	S	23	0-2	53
T	T	24	0-3	54
U	U	25	0-4	55
V	V	26	0-5	56
W	W	27	0-6	57
X	X	30	0-7	58
Y	Y	31	0-8	59
Z	Z	32	0-9	5A
0	0	33	0	30
1	1	34	1	31
2	2	35	2	32
3	3	36	3	33
4	4	37	4	34
5	5	40	5	35
6	6	41	6	36
7	7	42	7	37
8	8	43	8	38
9	9	44	9	39
+	+	45	12	2B
-	-	46	11	2D
*	*	47	11-8-4	2A
/	/	50	0-1	2F
$	$	53	11-8-3	24
=	=	54	8-3	3D
blank	blank	55	no punch	20
,(comma)	,(comma)	56	0-8-3	2C
.(period)	.(period)	57	12-8-3	2E

CMSE Error Messages C

CMSE Error Messages

The following error messages appear during CMSE operation.

Keyboard Error Line Messages

Message	Description
ADDRESS PARAMETER ERROR	An address parameter references a nonexistent address or an address that should not be modified.
BP3 IS WRITE-ONLY, CANNOT BE DISPLAYED	CYBER 990 memory BP3 cannot be displayed because it cannot be read.
CANNOT ADD TO MSL LIBRARY ON TAPE	When running from tape-based MSL, writing of data to tape is not allowed.
CANNOT CHANGE CM USAGE WITHOUT DEADSTART	An *RV command was rejected. An attempt was made to load overlays to CM with the CM usage option selected and a dedicated load device not present (AT511A/AT512A IOU only).
CANNOT DELETE FROM MSL LIBRARY ON TAPE	When running from tape-based MSL, deleting programs from the tape is not allowed.
CANNOT LOAD CW AND DOWNLOAD CW ON SAME CHANNEL	It is impossible for CMSE to download controlware over the same channel being used to read the controlware to be downloaded.
CHANNEL NUMBER NOT VALID	The specified channel does not exist.
CHANNEL XX IS RESERVED	The requested channel is reserved to a PP or CMSE monitor usage.
CM ADDRESS NOT VALID- CM LOCKED	An attempt was made to modify a CM word which is part of the CMSE area of CM that is in use.
CM LOAD NOT VALID- CM LOCKED	An attempt was made to load a program into CM which would have erased data in the CMSE area of CM that is in use.
CM NOT IN USE	A command was entered which affects use of the CMSE area of CM, such as UL, but CMSE is not using CM.
CM OVERLAYS CORRUPTED, REVERTING TO DISK	CMSE could not locate an overlay in CM. Overlays will be obtained from disk until an *OV command is entered (AT511A/AT512A IOU only).
CMSE COMMUNICATIONS ERROR	The CMSE monitor unexpectedly lost communications with the I/O driver or the display driver PP.

Message	Description
COMMAND BUFFER COMMAND NOT FOUND	A command buffer editing or execution command was entered which referenced a command number which could not be found.
COMMAND BUFFER MUST HAVE A NAME	A DE command was entered to delete command number zero, the command buffer name, which is not allowable unless a replacement name is provided.
COMMAND BUFFER TOO BIG FOR EDITING	An AE command was entered to edit a command buffer, but the command buffer was too big to fit into CBU's editing buffer space.
COMMAND NUMBER NOT VALID	A command buffer edit or display command referenced a command number which did not exist.
CPU NUMBER ERROR	A CPU number parameter on a command referred to a CPU which does not exist.
DATA PARAMETER ERROR	The entry for a data parameter was not in hexadecimal or octal format.
DAYFILE UNAVAILABLE FROM TAPE	An attempt was made to add to the dayfile via a command or call but adding to the dayfile is not allowed when MSL is being run from tape.
DELETING MSB IS NOT ALLOWABLE	Deleting MSB from MSL is not allowed because MSB is essential for deadstarting MSL. There is no possible way of patching MSB under CMSE, so there is no reason to want to delete MSB. MSB can be replaced, however, via the replace duplicate name featuring using TDX.
DISPLAY TRUNCATED-UNDISPLAYABLE DATA	The display of a command buffer was truncated because the command buffer contained data which was not possible to display.
ENTERING REG. 21 WHEN CM IN USE NOT ALLOWABLE	An ER command to enter data into the bounds register is not allowed when CM is in use because CMSE could hang if certain values were entered.
ERROR- NO COMMAND BUFFER NAME GIVEN	A command was entered which requires a name parameter but no name was given.
ERROR- NO SHADOW MEMORY AVAILABLE	A command was entered with shadow memory parameters, but no shadow memory exists.
ERROR OCCURRED WHILE IDLING CBU PP	An error occurred on the KE command while the CBU PP was being idled. The CBU PP should be manually deadstarted.
ERROR- PP MEMORY TRANSFER NOT COMPLETED	An error occurred while transferring data from one PP to another (MP command).

Message	Description
ERROR- 42 CALL TO LOAD TO NONEXISTENT CPU	An error occurred when the CPU could not be located.
FORMAT ERROR	There was an error in the sequence or content of the parameters of a command.
LINE NUMBER TOO BIG- 13H IS MAXIMUM	An AM command specified a line number which is not allowable.
MAINT CHAN REJECT	CMSE attempted to issue a function across the maintenance channel which was rejected.
MAINT CHAN RSVD	CMSE could not complete a request to do something which requires use of the maintenance channel, because it was reserved by a pool PP.
MOVE INTO LOCKED AREA OF CM NOT ALLOWABLE	An MC command was entered which attempted to move data into the CMSE area of CM that is in use.
MxP OVERLAY NOT FOUND IN CM	The CMSE monitor was unable to find an overlay in CM (CM in use or *OV command has been executed). A deadstart is likely to be required.
NAME DISPLAY UNAVAILABLE FROM TAPE	An AF or AG command was entered from an MSL tape device, and those displays are unavailable from tape.
NAME LOADED IN PPxx	This is an informative message which is displayed when a program is loaded into a PP, giving the name of the program loaded and the PP number.
NEW COMMAND CAUSES COMMAND BUFFER TO BE TOO BIG	A command buffer edit command was entered which increased the size of the command buffer and caused the command buffer to be too big for the buffer in CBU. The new data is not entered into the command buffer.
NO ACTIVE COMMAND BUFFER	An attempt was made to execute a command buffer from the command buffer editing utility, but no command buffer was being edited (CBU was not loaded).
NO COMMAND BUFFER AVAILABLE FOR EDITING	A command buffer edit command was entered, but no command buffer was being edited at the time.
NO COMMAND BUFFER AVAILABLE TO WRITE TO DISK	A *WB command was entered, but no command buffer was being edited at the time.
NO PP AVAILABLE TO LOAD COMMAND BUFFER EDITING UTILITY INTO	An AE command was entered to begin editing a command buffer, but CBU was not loaded, and there was no PP available to load CBU into.

Message	Description
NO PP AVAIL IN CLUSTER FOR CW LOAD	A PP is not available for CMSE in the cluster that has access to the channel. The PPs should be deadstarted and idle (AT511A/AT512A IOU only).
NO PP AVAIL IN CLUSTER FOR CHANNEL COMMAND	A PP is not available for CMSE in the cluster that has access to the channel. The PPs should be deadstarted and idle (AT511A/AT512A IOU only).
NO PP HAS BEEN AUTO ASSIGNED	A command was entered with a slash (/) in place of the PP number to use automatic PP assignment, but no PP had been automatically assigned with a PP load command.
NO PROGRAM NAME GIVEN	A command was entered which requires a program name parameter, but none was given on the command.
NOT ACCESSIBLE FROM IOU1	A memory or processor maintenance channel command was entered when running CMSE from IOU1 on a dual-IOU mainframe.
OVERLAYS ALREADY IN CM	A *OV command was entered to load monitor overlays into CM, but monitor overlays were already in CM because CM was in use by CMSE and the command is aborted.
PP COMMUNICATIONS ERROR	An error occurred in communications between the CMSE monitor and a pool PP.
PP UNAVAILABLE	An attempt was made to use a PP that does not exist or which is unavailable, because it is being used by CMSE or has been downed by a DN command.
PP xx LOAD ERROR	An error occurred in loading a program into PP xx.
PP xx NOT IN CONTACT WITH CMSE FOR DISPLAY REFRESH	This message appears above a PP memory display when the CMSE monitor and PP cannot communicate to refresh the memory display.
SOFT CONTROL MEMORY NAME ERROR	The parameter given for a soft control memory name was in error (model 990 only).
SOFT CONTROL MEMORY NUMBER ERROR	The parameter given for a soft control memory number was in error (models 840, 845, 850, 855, and 860 only).
TEST COMMAND	A command was entered which CMSE does not recognize as a command to CMSE, and therefore assumes it is a command to a test or diagnostic. This message displays until a test or diagnostic accepts the command from CMSE.
UNABLE TO CONTACT COMMAND BUFFER EDITING PP	A command buffer edit command failed because the CMSE monitor lost communications with the CBU PP.

Message	Description
UNABLE TO CONTACT PP	CMSE was unable to accomplish a task because CMSE was unable to establish communications with a PP, and these communications are necessary to accomplish the task.
VERIFY ERROR	An error was discovered when the data loaded to control store or control word via a VK or VT command was checked against the data that originated on the MSL device.
WARNING- KE COMMAND TERMINATED COMMAND BUFFER	This is a warning message which means that a KE command was entered to drop CBU, while a command buffer in CBU was being executed. The command buffer execution is terminated.
2 CALL PP LOAD ERROR	An error occurred in loading a program into a PP as a result of a 2 call.
8K BIT MUST BE SET	An ER command was entered to modify the IOU DEC Register (30H) on an AT478A/AT481A or AT478B/AT481B IOU with a data parameter which would have cleared the 8K bit and set the IOU into 4K mode. This is not allowable, because CMSE must execute in 8K mode.

CMSE Disk Driver Normal Running Messages

Message	Description
CLEARING CM	Appears during execution of the KC command when size of CM to be cleared is larger than one megabyte (MB). It appears once for each MB of CM cleared.
LOADING CMSE	Appears when CMSE is being loaded. It appears once for the monitor load, once for the display driver load, and once for each monitor overlay loaded to CM when CM is being used by CMSE.

CMSE Disk Driver Initial Display and Standalone Error Messages

Message	Description
CM ERROR	Appears when the driver detects an error during execution of the KC command.
ERROR CLEARING PP XX	Appears when the driver detects an error during execution of the KP command. XX is the number of the PP being cleared when the error is detected.
FORMAT ERROR	Appears when the syntax of an operator entry does not match the required format for the entry.
FUNCTION REJECT XXXX	Function code XXXX was issued to the MSL device and the correct response was not received.
GENERAL STATUS ERROR XXXX	Appears when a disk error is encountered while performing driver program loads or standalone program load operations. XXXX is the octal value of the general status word received. YYYY is the octal value of the contents of the detail status words received at the time of the error. Only a deadstart clears the message. The error message is displayed as follows:

```
GENERAL STATUS ERROR XXXX
WD      DETAIL STATUS
  1  YYYY   YYYY   YYYY   YYYY
  5  YYYY   YYYY   YYYY   YYYY
  9  YYYY   YYYY   YYYY   YYYY
 13  YYYY   YYYY   YYYY   YYYY
 17  YYYY   YYYY   YYYY   YYYY
```

Message	Description
INVALID ENTRY	Appears when the value entered for a parameter or an option is not within the legal limits for the selection.
I/O WORD COUNT ERROR	The A register of the driver PP contained a nonzero value at the assumed completion of an I/O operation to the MSL device by the driver. Only a deadstart clears the message.
MCH NOT RESPONDING	Appears if the driver is unable to input data from a maintenance register.

Message	Description
MR ERROR	Appears if an error condition is detected in the IOU status summary register. The message is cleared automatically when the error condition is cleared.
MSL BOUNDS ERROR	An attempt was made to read or write to an area of the MSL device outside of the area reserved for the MSL. Re-enter the last command to retry the operation. Only a deadstart clears the message.
PP EXCEEDS CLUSTER LIMIT	One of the PPs selected when the Monitor/Display Driver PP Numbers Command was entered is not in cluster zero of the IOU. Only PPs in cluster zero may be selected.
XXX NOT ON LIBRARY	Appears if a requested file name (XXX) was not found in the disk or dedicated load device. If the message comes from the library driver for a CMSE loader program, a deadstart is required to continue. If the message comes from the standalone loader, the message is presented in the keyboard entry line of the initial display, and the message can be cleared and normal input accomplished.
XXX NOT STAND-ALONE	Appears when a program binary that is requested to be loaded as a standalone program does not meet the standalone program criteria.

CMSE Disk Driver to CMSE Monitor Error Messages

Message	Description
CB READ ERROR	Indicates that the command buffer command count in the first sector was incorrect. The first command of the command buffer is not command number one. The command buffer must be recreated before it can be executed.
COMMAND BUFFER EMPTY	The command buffer that was requested to be loaded did not contain any data. It was a zero length file.
COMMAND BUFFER TO LARGE FOR EDITING	This message indicates that the command buffer that was requested for editing is greater than 3100 octal PP words.
COMMAND LENGTH EQUALS ZERO	A request to write a command buffer command of zero words was requested by CBU. This is not a valid request.
CONTROLWARE LOAD FAILURE	The attempt to load controlware on the channel specified in the command failed. Examine the command for the correct controlware name and I/O channel, make the necessary changes and retry the operation.
ERROR LOADING XXX	This message indicates that the program indicated by XXX could not be loaded due to a hardware failure or failure to initialize the hardware correctly.
FILE NOT AVAILABLE	Indicates an attempt was made to write data into a closed file. The file must be deleted from the library and recreated as an open file before data can be added to it.
FUNCTION REJECT XXXX	The function code XXXX was issued to the MSL device and the correct response was not received. Re-enter the last command to retry the operation.
GENERAL STATUS ERROR XXXX (FSEC=YYYY/CSDEC=ZZ)	An error code XXXX was encountered by the driver during an MSL operation. The CC598A error code FSEC is indicated by YYYY and CSDEC error code by ZZ. These error codes are defined in the CYBER Systems Peripheral Diagnostic Reference Manual listed in About This Manual. Retry by re-entering the command.
I/O WORD COUNT ERROR	The A register of the driver PP contained a nonzero value at the assumed completion of an I/O operation to the MSL device by the driver.

Message	Description
LOAD ABORT - MEMORY BUFFER OVERFLOW	This message occurs if one of two conditions exist. The first condition is if an attempt is made to load a program into a PP that overwrites the monitor to a pool PP communication block in the upper memory of the PP. The second condition is if an attempt is made by the driver to write data to CM in the area reserved by CMSE when CM usage has been enabled. Retry for the first condition is to load the program to CM instead of a PP. Retry for the second condition is to do one of the following: • Deadstart, clear the CM usage parameter on the initial display, reload CMSE, and retry the operation. • Attempt to set the address parameter on the command to load at an address that does not cause overflow.
MSL BOUNDS ERROR	An attempt was made to read or write to an area of the MSL device outside of the area reserved for the MSL. Re-enter the last command to retry the operation.
PNT FULL	A program or command buffer cannot be added to the MSL for lack of an open entry in the PNT. Examine the PNT for an unnecessary program or command buffer. If found, delete the unnecessary program or command buffer and retry the operation.
PRFX TABLE MISMATCH	This message occurs if the name in the PRFX table (77 table) that the I/O driver just read does not match the program name requested. If CM is being used by CMSE, this error causes the CM directory to be invalidated. **NOTE** Deleting any DEMOT created file without a 77 table (for example, Flaw map from FMU) results in the error message PRFX TABLE MISMATCH. Although the file was deleted, the disk space used by the file is not released by CMSE. If an SRT FULL situation is encountered, CIP requires reinstallation using the UPDATE installation option.
SRT FULL	A program or command buffer cannot be added to the MSL for lack of available storage space on the MSL disk. Examine the PNT for an unnecessary program or command buffer. If found, delete the unnecessary program or command buffer and retry the operation.

Message	Description
TO MANY CHARACTERS IN THE COMMAND LINE	There were more than 64 decimal characters in a command buffer command line.
XXX DUPLICATE NAME	The program or command buffer name XXX currently exists in the MSL PNT. Examine the name and change it if necessary or delete the existing program or command buffer of the same name and retry the operation.
XXX IS A HEX PROGRAM	An attempt was made to load a hexadecimal program as a standalone program on a model 865 or 875.
XXX NOT ON LIBRARY	The program or command buffer name XXX (one to seven alphanumeric characters) was not found on the MSL. Check the program or command buffer name and retry the command.

IOU Error Messages

The following error messages may appear on the bottom line of the A display.

Message	Description
IOU ERO - ADU - B	Parity error detected from barrel priority ROM.
IOU ERO - A/R	Error detected on the A/R register pack.
IOU ERO - CHxx	Parity error on channel xx.
IOU ERO - CM AD - O	Parity error was detected on the output of a CM address.
IOU ERO - CM DA-IN	Parity error detected on the input of CM data.
IOU ERO - CM DA - O	Parity error was detected on the output of CM data.
IOU ERO - CM IN BYxx	Parity error in CM data-in byte xx.
IOU ERO - CM OUT BYxx	Error in CM data-out byte xx.
IOU ERO - CM RESP	CM response code error.
IOU ERO - FMWR	Control firmware has detected a parity error.
IOU ERO - M ADV	Error detected on PPM data in the ADV pack.
IOU ERO - OSB A	Parity error detected on the OS boundary address.
IOU ERO - OSB V	PP has attempted to write or exchange at a CM address outside the region allowed by the OS bounds register.
IOU ERO - PP AD	Address parity error has been detected on the memory pack.
IOU ERO - PP IN	Parity error has been detected on the memory pack.
IOU ERO - PP OUT	Parity error has been detected on data read from PP memory.
IOU ERO - P/Q	Error condition detected on the P/Q register pack.
IOU ERO - PPxx	Error in PP xx.
IOU ERO - RD	Uncorrected CM read error.
IOU ERO - RD BF	Read data from CM has a parity error.
IOU ERO - REJ	CM reject error.
IOU ERO - TAG IN	Tag from CM has a parity error.
IOU ERO - WRT	Uncorrected CM write error.
IOU ERO - 12/16 CNV	Error detected on the 12/16 conversion pack.
IOU ERO - 7xxx	Error condition has been detected on a 7xxx pack. xxx is the pack type.

Memory Error Messages

Message	Description
`Mx ERO-BOUNDS FAULT`	Write operation has exceeded the address bounds as specified by the bounds register. x — System number (1, 2, 3, or 5).
`Mx ERO-INVALID FNC`	Invalid function has been decoded. x — System number (1, 2, 3, or 5).
`Mx ERO-Mm Lv PTp CAaa RArr BKb PRt BYy DIii MPxx FPff`	Uncorrected Mx error 1: x — System number 1, 2, or 3 (models 845 and 855 only). m — 0 - No multibit error. 1 - Multibit error. v — 1 - Error at the memory port. 2 - Parity error at the distributor or array pack. p — Memory port number (0 through 3). aa — Chip column address (0 through 7F). rr — Chip row address (0 through 1F). b — Memory bank number (0 through F). t — Parity bit number (0 through 7). y — Parity error byte position code (0 through F). ii — Data in parity bits (0 through FF). xx — Mark and parity bits (0 through FF). ff — Function and parity code (0 through FF).
`Mx ERO-PORTp CAaa RArr BNKb PARt SYNDss`	Corrected Mx error: x — System number 1, 2, or 3 (models 845 and 855 only). p — Memory port number (0 through 7). aa — Chip column address (0 through 7F). rr — Chip row address (0 through 1F). b — Memory bank number (0 through 7). t — Parity bit number (0 through 7). ss — Syndrome bits (0 through FF).

Message	Description
`Mx ERO-TAGg PRTp CAaa RArr BNKb PARt DPdd`	Uncorrected Mx error 2:

x	System number (1, 2, or 3).
g	1 - Parity error in the 8-bit tag returned to the processor.
p	Memory port number (0 through 7).
aa	Chip column address (0 through 7F).
rr	Chip row address (0 through 1F).
b	Memory bank number (0 through 7).
t	Parity bit number (0 through 7).
dd	Data out path parity error when DP is displayed, or partial write parity byte (0 through 7) if PW is displayed.

Message	Description
`M3 ERO-DPE7b PORTp ADaaaaaaa PRTYt RDPrr PWDPww (CYBER 960 only)`	Uncorrected M3 error 2:

b	Data parity error byte 7 (0 and 1).
p	Memory port number (0 through 7).
a	Address (0 through FFFFFFF).
t	Parity bit number (0 through F).
rr	Read data parity (0 through FF).
ww	Partial write data parity (0 through FF).

NOTE

The following entire message is too long to display during log errors operation. Only the first portion through DISTd is displayed, but the entire message is logged.

Message	Description
`M3 ERO-Mm Lv PTp ADaaaaaaa PRTYt BYy DIii MPxxx FPff (models 850 and 860 only)`	Uncorrected M3 error 1:

m	0 - No multibit error. 1 - Multibit error.
v	1 - Error at the memory port. 2 - Parity error at the distributor or array pack.
p	Memory port number (0 through 7).
a	Address (0 through 3FFFFFF).
t	Parity bit number (0 through F).
y	Parity error byte position code (0 through F).
ii	Data in parity bits (0 through FF).
xxx	Mark bits and parity (0 through 1FF).
ff	Function bits and parity (0 through 1F).

Message	Description
M3 ERO-Mm PTp ADaaaaaaa PRTYt BYy DIii MPxxx FPff (CYBER 960 only)	Uncorrected M3 error 1: m — 0 - No multibit error. 1 - Multibit error. p — Memory port number (0 through 7). a — Address (0 through FFFFFFF). t — Parity bit number (0 through F). y — Parity error byte position code (0 through F). ii — Data in parity bits (0 through FF). xxx — Mark bits and parity (0 through 1FF). ff — Function bits and parity (0 through 1F).
M3 ERO-PORTp ADaaaaaaa PRTYt RDPrr PWDPww (models 850 and 860 only)	Uncorrected M3 error 2: p — Memory port number (0 through 7). a — Address (0 through 3FFFFFF). t — Parity bit number (0 through F). rr — Read data parity (0 through FF). ww — Partial write data parity (0 through FF).
M3 ERO-PORTp ADaaaaaaa PRTYt SYNDss	Corrected M3 error:[1] p — Memory port number (0 through 7). a — Address (0 through 3FFFFFF for models 850 and 860 and 0 through FFFFFFF for model 960). t — Address parity (0 through F). s — Syndrome bits (0 through FF).

1. CYBER models 850, 860, and CYBER 960 only.

Message	Description
`M5 ERO-Ax IFi BFf Mm PAe CSu PTp COLc RSr CAaaaa BNKb DISTd PARt WDww PRWyy APEz MTFPv (model 990 only)`	Uncorrected M5 error:

x	Register number (4, 5, 6 or 7).
i	Illegal function (0 or 1).
f	Bounds fault (0 or 1).
m	Multi bit error (0 or 1).
e	Port address PE (0 or 1).
u	CSU PE (0 or 1).
p	Memory port number (0 through 7).
c	CSU column number (0 or 1).
r	Chip row select (0 through 3).
aaaa	Chip address (0 through FFFF).
b	Memory bank number (0 through 7).
d	Cage (distributer) select (0 through 3).
t	Address parity (0 through F).
ww	Write data PE (0 through FF).
yy	Read or partial write PE (0 through FF).
z	Address PE (0 through F).
v	Mark PE, tag PE, function PE, or partial write PE (0 through F).

Message	Description
`M5 ERO-Ax PTp COLc RSr CAaaaa BNKb DISTd PARt SYNDss`	Corrected M5 error:[2]

x	Register number (0, 1, 2, or 3).
p	Memory port number (0 through 7).
c	CSU column number (0 through 1).
r	Chip row select (0 through 3).
aaaa	Chip address (0 through FFFF).
b	Memory bank number (0 through 7).
d	Cage (distributor) select (0 through 3).
t	Address parity (0 through F).
ss	Syndrome bits (0 through FF).

2. CYBER model 990 only.

Px Error Messages

Message	Description
Px CPy ER-xxx PFS-ssssssssssssssss CEL-cccccccccccccccc	x System number (1 or 2). y Number of CPU in which error occurred. xxx UCE - Uncorrected error. CRE - Corrected error. PEW - Physical environment warning. s Processor fault status. c CACHE corrected error log.
Px CPy ER-UCE[3] 8n-ssssssssssssssss[3] 8n-ssssssssssssssss[3]	x System number (3 or 5). y Number of CPU in which error occurred. n Processor fault status register number. s Processor fault status.

3. CYBER models 840, 845, 850, 855, 860, and 990 only.

MSL Library D

Tables within this appendix indicate which test, diagnostic, and support programs constitute the MSL library for the following computer systems:

MSL Library	Associated Computer Systems
MSL151	Models 810/815, and 825/830
MSL152	Model 835
MSL153	Models 840, 845/850, and 855/860/870
CYBER 96X	CYBER 960 and 962
MSL 155	Models 990/995, CYBER 992 and 994

An 'X' in the right hand column of a table indicates the existence of a particular program within an MSL library, while a '-' indicates the absense of a program.

Refer to the appropriate reference manual or microfiche listings of individual tests for a more detailed description.

COMMON TEST AND INITIALIZATION		MSL 151		MSL 152	MSL 153			CYBER 96X		MSL 155		
		CIP 810/ 815	CIP 825/ 830	CIP 835	CIP 840	CIP 845/ 850	CIP 855/ 860/ 870	CIP 960	CIP 962	CIP 990/ 995	CIP 992	CIP 994
CTI	Common test and initialization (includes IPL - ZZZ)	X	X	X	X	X	X	X	X	X	X	X

COMMON MAINTENANCE SOFTWARE EXECUTIVE		MSL 151		MSL 152	MSL 153			CYBER 96X		MSL 155		
		CIP 810/ 815	CIP 825/ 830	CIP 835	CIP 840	CIP 845/ 850	CIP 855/ 860/ 870	CIP 960	CIP 962	CIP 990/ 995	CIP 992	CIP 994
BIF	CMSE information display	X	X	X	X	X	X	-	-	-	-	-
BIF4	CMSE information display	-	-	-	X	X	X	X	X	X	X	X
DSB	HIVS disk bootstrap loader	X	X	X	X	X	X	-	-	X	-	X
DSD	Dual screen display driver, CC598A/B with advanced emulation package	-	-	-	-	-	-	X	X	X	X	X
DSL8XX	MDL standalone loader	X	X	X	X	X	X	X	-	X	-	X
DSP	CC545 Display driver	X	X	X	X	X	X	X	-	X	-	X
DSQ	CC634B Display driver	X	X	X	X	X	X	X	-	X	-	X
DSQ4	CC634B & CC598A/B Display driver	-	-	-	X	X	X	X	X	X	X	X
KDP8XX	Keyboard/display processor	X	X	X	X	X	X	X	-	X	-	X
MDL	Maintenance disk loader	X	X	X	X	X	X	X	-	X	-	X
MDL1	844 Disk driver	X	X	X	X	X	X	X	-	X	-	X
MDL2	885 Disk driver	X	X	X	X	X	X	X	-	X	-	X
MDL3	834 Disk driver	X	X	-	-	-	-	-	-	-	-	-
MDL4	895 Disk driver	-	-	X	X	X	X	X	-	X	-	X

(continued)

		MSL 151		MSL 152	MSL 153			CYBER 96X		MSL 155		
COMMON MAINTENANCE SOFTWARE EXECUTIVE *(continued)*		CIP 810/ 815	CIP 825/ 830	CIP 835	CIP 840	CIP 845/ 850	CIP 855/ 860/ 870	CIP 960	CIP 962	CIP 990/ 995	CIP 992	CIP 994
MDL5	836 Disk driver	X	X	-	-	-	-	-	-	-	-	-
MDLD	Dedicated load device loader	-	-	-	-	-	-	-	X	-	X	-
MDP	CMSE monitor/executive	X	X	-	-	-	-	-	-	-	-	-
MEP	CMSE monitor/executive	-	-	X	-	-	-	-	-	-	-	-
MFP	CMSE monitor/executive	-	-	-	X	X	X	-	-	-	-	-
MF4	CMSE monitor/executive	-	-	-	X	X	X	-	-	-	-	-
MG4	CMSE monitor/executive	-	-	-	-	-	-	X	-	-	-	-
MG5	CMSE monitor/executive	-	-	-	-	-	-	X	X	-	-	-
MH4	CMSE monitor/executive	-	-	-	-	-	-	-	-	X	-	X
MH5	CMSE monitor/executive	-	-	-	-	-	-	-	-	X	X	X
MID	Keyboard/display processor	-	-	-	-	-	-	-	X	-	X	-
MSB	CMSE disk bootstrap loader	X	X	X	X	X	X	X	-	X	-	X
TDL	CMSE tape loader	X	X	X	X	X	X	X	-	X	-	X
TDLCRES	Tape driver channel resident	X	X	X	X	X	X	X	-	X	-	X
TDLMRES	Tape driver memory resident	X	X	X	X	X	X	X	-	X	-	X
TLO	Standalone loader	X	X	X	X	X	X	X	-	X	-	X
TSL8XX	Standalone loader	X	X	X	X	X	X	X	-	X	-	X

UTILITY PROGRAMS		MSL 151		MSL 152	MSL 153			CYBER 96X		MSL 155		
		CIP 810/ 815	CIP 825/ 830	CIP 835	CIP 840	CIP 845/ 850	CIP 855/ 860/ 870	CIP 960	CIP 962	CIP 990/ 995	CIP 992	CIP 994
ABU	Assembly buffer utility	X	X	X	X	X	X	-	-	-	-	-
ABUP	Assembly buffer utility	-	-	-	-	-	-	X	X	X	X	X
CBU	Command buffer utility	X	X	X	X	X	X	X	X	X	X	X
CBD	Capture buffer formatter	-	-	-	-	-	-	-	-	X	-	X
CLK	Real time clock	X	X	X	X	X	X	X	X	X	X	X
DMP	PP and CM line printer dump	X	X	X	X	X	X	X	-	X	-	X
DDS	Deadstart diagnostic sequencer	X	X	X	X	X	X	-	-	X	-	X
EME	Establish maintenance environment	X	X	X	X	X	X	X	X	X	X	X
EXC	CPU executive	X	X	X	X	X	X	X	X	X	X	X
ITL4	IPC EEPROM loader	-	-	-	X	X	X	X	X	X	X	X
LEEP	CM3 EEPROM loader	-	-	-	X	X	X	X	X	X	X	X
TDX	MSL tape-to-disk loader	X	X	X	X	X	X	X	-	X	-	X
UTCM	Clear CM utility	-	-	-	X	X	X	X	X	X	X	X

IOU TESTS		MSL 151		MSL 152	MSL 153			CYBER 96X		MSL 155		
		CIP 810/ 815	CIP 825/ 830	CIP 835	CIP 840	CIP 845/ 850	CIP 855/ 860/ 870	CIP 960	CIP 962	CIP 990/ 995	CIP 992	CIP 994
CCA4	CYBER 170 CH adapter	-	-	-	X	X	X	X	-	X	-	X
CCA44	DMA enhanced CYBER CH adapter	-	-	-	-	-	-	X	X	X	X	X
CHD1	Channel test	X	X	-	-	-	-	-	-	-	-	-
CHD2	Channel test	-	-	X	X	X	X	-	-	-	-	-
CHD4	Channel test	-	-	-	X	X	X	X	-	X	-	X
CHD44	Channel test	-	-	-	-	-	-	X	X	X	X	X
CMA1	Central memory access test	X	X	-	-	-	-	-	-	-	-	-
CMA2	Central memory access test	-	-	X	X	X	X	-	-	-	-	-
CMA4	Central memory access test	-	-	-	X	X	X	X	-	X	-	X
CMA44	Central memory access test	-	-	-	-	-	-	X	X	X	X	X
CRA1	Clock & remote access test	X	X	-	-	-	-	-	-	-	-	-
CRA4	Clock & remote access test	-	-	-	X	X	X	X	-	X	-	X
CSC1	Central storage confilct test	X	X	-	-	-	-	-	-	-	-	-
DST1	Display alignment test	X	X	-	-	-	-	-	-	-	-	-
DST2	Display alignment test	-	-	X	X	X	X	-	-	-	-	-
DST4	Display alignment test	-	-	-	X	X	X	X	X	X	X	X
EDS1	Extended deadstart test	X	X	-	-	-	-	-	-	-	-	-
EDS2	Extended deadstart test	-	-	X	X	X	X	-	-	-	-	-
EDS4	Extended deadstart test	-	-	-	X	X	X	X	-	X	-	X
EXT1	Executive unit test	X	X	-	-	-	-	-	-	-	-	-
EXT2	Executive unit test	-	-	X	X	X	X	-	-	-	-	-
EXT4	Executive unit test	-	-	-	X	X	X	X	-	X	-	X
EXT44	Executive unit test	-	-	-	-	-	-	X	X	X	X	X

(continued)

MSL Library

		MSL 151		MSL 152	MSL 153			CYBER 96X		MSL 155		
IOU TESTS *(continued)*		CIP 810/ 815	CIP 825/ 830	CIP 835	CIP 840	CIP 845/ 850	CIP 855/ 860/ 870	CIP 960	CIP 962	CIP 990/ 995	CIP 992	CIP 994
FII1	IOU fault isolation program	X	X	-	-	-	-	-	-	-	-	-
FII2	IOU fault isolation program	-	-	X	X	X	X	-	-	-	-	-
FII4	IOU fault isolation program	-	-	-	X	X	X	X	-	X	-	X
FII44	IOU fault isolation program	-	-	-	-	-	-	X	X	X	X	X
IPI4	IPI channel adapter test	-	-	-	X	X	X	X	-	X	-	X
IPI44	IPI channel adapter test	-	-	-	-	-	-	X	X	X	X	X
ISI4	ISI channel adapter test	-	-	-	X	X	X	X	-	X	-	X
ISI44	ISI channel adapter test	-	-	-	-	-	-	X	X	X	X	X
MRA1	Maintenance register test	X	X	-	-	-	-	-	-	-	-	-
MRA2	Maintenance register test	-	-	X	X	X	X	-	-	-	-	-
MRA4	Maintenance register test	-	-	-	X	X	X	X	-	X	-	X
MRA44	Maintenance register test	-	-	-	-	-	-	X	X	X	X	X
MRC1	Maintenance register test	X	X	-	-	-	-	-	-	-	-	-
MRTC	Maintenance register test	X	X	-	-	-	-	-	-	-	-	-
MRT1	Maintenance register test	X	X	-	-	-	-	-	-	-	-	-
MRT2	Maintenance register test	-	-	X	X	X	X	-	-	-	-	-
MRT4	Maintenance register test	-	-	-	X	X	X	X	-	X	-	X
MRT44	Maintenance register test	-	-	-	-	-	-	X	X	X	X	X
MUX1	Two port MUX test	X	X	-	-	-	-	-	-	-	-	-
PMT1	PP memory test one	X	X	-	-	-	-	-	-	-	-	-
PMT2	PP memory test one	X	X	X	X	X	X	-	-	-	-	-
PMT4	PP memory test one	-	-	-	X	X	X	X	-	X	-	X
PMT44	PP memory test one	-	-	-	-	-	-	X	X	X	X	X

(continued)

IOU TESTS *(continued)*		MSL 151		MSL 152	MSL 153			CYBER 96X		MSL 155		
		CIP 810/ 815	CIP 825/ 830	CIP 835	CIP 840	CIP 845/ 850	CIP 855/ 860/ 870	CIP 960	CIP 962	CIP 990/ 995	CIP 992	CIP 994
PMU1	PP memory test two	X	X	-	-	-	-	-	-	-	-	-
PMU2	PP memory test two	-	-	X	X	X	X	-	-	-	-	-
PMU4	PP memory test two	-	-	-	X	X	X	X	-	X	-	X
PMU44	PP memory test two	-	-	-	-	-	-	X	X	X	X	X
QLT1	Quick look test	X	X	-	-	-	-	-	-	-	-	-
QLT2	Quick look test	-	-	X	X	X	X	-	-	-	-	-
QLT4	Quick look test	-	-	-	X	X	X	X	-	X	-	X
QLT44	Quick look test	-	-	-	-	-	-	X	X	X	X	X
TPA4	Two port MUX test	-	-	-	-	-	-	X	X	-	X	X
TPM1	Two port MUX test	X	X	-	-	-	-	-	-	-	-	-
TPM2	Two port MUX test	-	-	X	X	X	X	-	-	-	-	-
TPM4	Two port MUX test	-	-	-	X	X	X	X	-	X	-	X
TPM44	Two port MUX test	-	-	-	-	-	-	X	X	X	X	X

CENTRAL MEMORY TESTS		MSL 151 CIP 810/815	MSL 151 CIP 825/830	MSL 152 CIP 835	MSL 153 CIP 840	MSL 153 CIP 845/850	MSL 153 CIP 855/860/870	CYBER 96X CIP 960	CYBER 96X CIP 962	MSL 155 CIP 990/995	MSL 155 CIP 992	MSL 155 CIP 994
CMI1	IOU-based CM isolation	X	X	-	-	-	-	-	-	-	-	-
CMI2	IOU-based CM isolation	-	-	X	-	-	-	-	-	-	-	-
CMI3	IOU-based CM isolation (CP845/855 BS137A memory)	-	-	-	X	X	X	-	-	-	-	-
CMJ3	IOU-based CM isolation (CP840/850/860/870 BS213A memory)	-	-	-	X	X	X	-	-	-	-	-
CMK3	IOU-based CM isolation	-	-	-	-	-	-	X	X	-	-	-
CMT1	IOU-based CM test	X	X	-	-	-	-	-	-	-	-	-
CMT2	IOU-based CM test	-	-	X	-	-	-	-	-	-	-	-
CMT3	IOU-based CM test	-	-	-	X	X	X	-	-	-	-	-
CMT3P	IOU-based CM test	-	-	-	-	-	-	X	X	-	-	-
CMT5	IOU-based CM test	-	-	-	-	-	-	-	-	X	X	X

PROCESSOR DETECTION & ISOLATION TESTS		MSL 151		MSL 152	MSL 153			CYBER 96X		MSL 155		
		CIP 810/ 815	CIP 825/ 830	CIP 835	CIP 840	CIP 845/ 850	CIP 855/ 860/ 870	CIP 960	CIP 962	CIP 990/ 995	CIP 992	CIP 994
ACT3	Address control test	-	-	-	X	X	X	-	-	-	-	-
ACT3P	Address control test	-	-	-	-	-	-	X	X	-	-	-
ACT5	Address control test	-	-	-	-	-	-	-	-	X	X	X
ANT3	Arithmetic & logic unit test	-	-	-	X	X	X	-	-	-	-	-
ANT3P	Arithmetic & logic unit test	-	-	-	-	-	-	X	X	-	-	-
BPT3	Business data processor test	-	-	-	X	X	X	-	-	-	-	-
BPT3P	Business data processor test	-	-	-	-	-	-	X	X	-	-	-
BPT5	Business data processor test	-	-	-	-	-	-	-	-	X	X	X
CST3	Control store basic path test	-	-	-	X	X	X	-	-	-	-	-
CST3P	Control store basic path test	-	-	-	-	-	-	X	X	-	-	-
CTT3	Common CST/MAC logic test	-	-	-	X	X	X	-	-	-	-	-
CTT3P	Common CST/MAC logic test	-	-	-	-	-	-	X	X	-	-	-
EPT5	Error processing network test	-	-	-	-	-	-	-	-	X	X	X
FIS1	Fault isolation test	X	X	-	-	-	-	-	-	-	-	-
FIS2	Fault isolation test	-	-	X	-	-	-	-	-	-	-	-
FPT5	Floating point unit test	-	-	-	-	-	-	-	-	X	X	X
ICT3	Instruction compete control, instruction control pipeline test	-	-	-	X	X	X	-	-	-	-	-
ICT3P	Instruction complete control, instructin control pipeline test	-	-	-	-	-	-	X	X	-	-	-
IDT5	Instruction decode test	-	-	-	-	-	-	-	-	X	X	X
IFT3	Instruction fetch test	-	-	-	X	X	X	-	-	-	-	-
IFT3P	Instruction fetch test	-	-	-	-	-	-	X	X	-	-	-
IFT5	Instruction fetch test	-	-	-	-	-	-	-	-	X	X	X
IGT5	Integer unit test	-	-	-	-	-	-	-	-	X	X	X

(continued)

PROCESSOR DETECTION & ISOLATION TESTS *(continued)*		MSL 151		MSL 152	MSL 153			CYBER 96X		MSL 155		
		CIP 810/815	CIP 825/830	CIP 835	CIP 840	CIP 845/850	CIP 855/860/870	CIP 960	CIP 962	CIP 990/995	CIP 992	CIP 994
LMT3	Local memory test	-	-	-	X	X	X	-	-	-	-	-
LMT3P	Local memory test	-	-	-	-	-	-	X	X	-	-	-
LST5	Load/Store unit test	-	-	-	-	-	-	-	-	X	X	X
MAT3	Memory array identification test	-	-	-	X	X	X	-	-	-	-	-
MAT3P	Memory array identification test	-	-	-	-	-	-	X	X	-	-	-
MAT5	Maintenance access control test	-	-	-	-	-	-	-	-	X	X	X
MCT3	Maintenance access control test	-	-	-	X	X	X	-	-	-	-	-
MCT3P	Maintenance access control test	-	-	-	-	-	-	X	X	-	-	-
MDT3	ALN multiply/divide test	-	-	-	X	X	X	-	-	-	-	-
MDT3P	ALN multiply/divide test	-	-	-	-	-	-	X	X	-	-	-
MIT5	Miscellaneous test	-	-	-	-	-	-	-	-	X	X	X
MST5	Memory stress test	-	-	-	-	-	-	-	-	X	X	X
OIT3	Operand issue test	-	-	-	X	X	X	-	-	-	-	-
OIT3P	Operand issue test	-	-	-	-	-	-	X	X	-	-	-
OCT5	Operand CACHE test	-	-	-	-	-	-	-	-	X	X	X
PAT5	Basic paths detection test	-	-	-	-	-	-	-	-	X	X	X
PDT3	Processor detected logic test and associated logic test	-	-	-	X	X	X	-	-	-	-	-
PDT3P	Processor detected logic test and associated logic test	-	-	-	-	-	-	X	X	-	-	-
SCT5	Shift unit detection test	-	-	-	-	-	-	-	-	X	X	X
SMT3	Segment MAP test	-	-	-	X	X	X	-	-	-	-	-
SMT3P	Segment MAP test	-	-	-	-	-	-	X	X	-	-	-
VAT5	Virtual addressing test	-	-	-	-	-	-	-	-	X	X	X

MODEL INDEPENDENT TESTS		MSL 151 CIP 810/815	MSL 151 CIP 825/830	MSL 152 CIP 835	MSL 153 CIP 840	MSL 153 CIP 845/850	MSL 153 CIP 855/860/870	CYBER 96X CIP 960	CYBER 96X CIP 962	MSL 155 CIP 990/995	MSL 155 CIP 992	MSL 155 CIP 994
BIMM2	BDP immediate endcase	-	-	-	X	X	X	X	X	X	X	X
BRCH2	Floating point branch endcase	-	-	-	X	X	X	X	X	X	X	X
BYTE2	BDP byte endcase	-	-	-	X	X	X	X	X	X	X	X
CACH2	Central memory test	-	-	X	X	X	X	X	X	X	X	X
CMEM1	Central memory test	X	X	-	-	-	-	-	-	-	-	-
CMEM2	Central memory test	X	X	X	X	X	X	X	X	X	X	X
CRPT2	Call/Return/Pop test	-	-	-	X	X	X	X	X	X	X	X
DBUG2	DEBUG test	X	X	-	X	X	X	X	X	X	X	X
DOBL2	Double precision FP endcase	-	-	-	X	X	X	X	X	X	X	X
EDIT2	BDP edit endcase	-	-	-	X	X	X	X	X	X	X	X
FCT1	Fixed operand command test	X	X	X	X	X	X	X	X	X	X	X
FCT2	Fixed operand command test	X	X	X	X	X	X	X	X	-	-	-
FCT3	Fixed operand command test	X	X	X	X	X	X	X	X	-	-	-
FCT5	Fixed operand command test	X	X	X	X	X	X	X	X	X	X	X
FCT9	Fixed operand command test	X	X	X	X	X	X	X	X	X	X	X
EXCH	Exchange test	X	X	X	X	X	X	X	X	X	X	X
FIMM2	Full word immediate endcase	-	-	-	X	X	X	X	X	X	X	X
FINT2	Full word integer endcase	X	X	X	X	X	X	X	X	X	X	X
HIMM2	Half word immediate endcase	-	-	-	X	X	X	X	X	X	X	X
HINT2	Half word integer endcase	-	-	-	X	X	X	X	X	X	X	X
KYPT2	Keypoint test	X	X	X	X	X	X	X	X	X	X	X
NUMR2	BDP numeric endcase	-	-	-	X	X	X	X	X	X	X	X
PAGE2	Page fault endcase test	-	-	-	X	X	X	X	X	X	X	X

(continued)

MODEL INDEPENDENT TESTS (continued)		MSL 151 CIP 810/815	MSL 151 CIP 825/830	MSL 152 CIP 835	MSL 153 CIP 840	MSL 153 CIP 845/850	MSL 153 CIP 855/860/870	CYBER 96X CIP 960	CYBER 96X CIP 962	MSL 155 CIP 990/995	MSL 155 CIP 992	MSL 155 CIP 994
RFST2	Random fast/slow test	X	X	X	X	X	X	X	X	X	X	X
RNGT2	Ring test	-	-	-	X	X	X	X	X	X	X	X
SNGL2	Single precision FP endcase	-	-	-	X	X	X	X	X	X	X	X
RCT12	Random command test	X	X	X	X	X	X	X	X	X	X	X
RCT22	Random command test	X	X	X	X	X	X	X	X	X	X	X
STAT2	CPYXS CPYSX CPYTX endcase	-	-	-	X	X	X	X	X	-	-	-
TASE2	Address spec error endcases test	-	-	-	X	X	X	X	X	-	-	-
TIVE2	Invalid segment endcase test	-	-	-	X	X	X	X	X	-	-	-
TRAP	Trap interrupt test	X	X	X	X	X	X	X	X	X	X	X
VCMP2	Vector compare endcase	-	-	-	-	-	-	-	-	X	X	X
VFLT2	Vector FP endcase	-	-	-	-	-	-	-	-	X	X	X
VGTH2	Vector gather/scatter endcase	-	-	-	-	-	-	-	-	X	X	X
VINT2	Vector integer endcase	-	-	-	-	-	-	-	-	X	X	X

		MSL 151		MSL 152	MSL 153			CYBER 96X		MSL 155		
CYBER 170 TESTS		CIP 810/ 815	CIP 825/ 830	CIP 835	CIP 840	CIP 845/ 850	CIP 855/ 860/ 870	CIP 960	CIP 962	CIP 990/ 995	CIP 992	CIP 994
ALX	Random CPU instruction test	X	X	X	X	X	X	X	X	X	X	X
BGK	30-Bit instruction test	X	X	X	X	X	X	X	X	X	X	X
CMC	Central memory conflict test	X	X	X	X	X	X	X	X	X	X	X
CM6	Central memory test	X	X	X	X	X	X	X	X	X	X	X
CT8	Random instruction test	X	X	X	X	X	X	X	X	X	X	X
CU8	CPU command test	X	X	X	X	X	X	X	X	X	X	X
EJP	GO - NO/GO exchange jump test	X	X	X	X	X	X	X	X	X	X	X
FDT	Floating divide test	X	X	X	X	X	X	X	X	X	X	X
FM2	Floating muliply tet	X	X	X	X	X	X	X	X	X	X	X
FS8	Random command test	X	X	X	X	X	X	X	X	X	X	X
IMC	Integer multiply test	X	X	X	X	X	X	X	X	X	X	X
LAT	Long add unit test	X	X	X	X	X	X	X	X	X	X	X
MY1	CPU test of central memory	X	X	X	X	X	X	X	X	X	X	X
POP	Population counter test	X	X	X	X	X	X	X	X	X	X	X
RTJ	Return jump test	X	X	X	X	X	X	X	X	X	X	X

		MSL 151		MSL 152	MSL 153			CYBER 96X		MSL 155		
COMPASS PERIPHERAL DIAGNOSTICS		CIP 810/ 815	CIP 825/ 830	CIP 835	CIP 840	CIP 845/ 850	CIP 855/ 860/ 870	CIP 960	CIP 962	CIP 990/ 995	CIP 992	CIP 994
BCX	BC Command test	X	X	X	X	X	X	X	-	X	-	X
CD8	66X/844 Coupler diagnostic	X	X	X	X	X	X	X	-	X	-	X
CID	BC Coupler isolation diagnostic	X	X	X	X	X	X	X	-	X	-	X
CP8	415 Card punch test	X	X	X	X	X	X	X	-	X	-	X
CR8	405 Card reader test	X	X	X	X	X	X	X	-	X	-	X
DPDI4	9836/9853 Disk subsystem test	-	-	-	X	X	X	X	X	X	X	X
DPDS	834/836 Disk data path integrity	X	X	-	-	-	-	-	-	-	-	-
DTA	844 Disk diagnostic	X	X	X	X	X	X	X	-	X	-	X
DTB	7X54 Diagnostic	X	X	X	X	X	X	X	-	X	-	X
ESMU	ESM utility/error log monitor	X	X	X	X	X	X	X	X	X	X	X
FM8	844/7X5X Disk pack formatting	X	X	X	X	X	X	X	-	X	-	X
FT8	7054 Full-track coupler test	X	X	X	X	X	X	X	-	X	-	X
HYDR	I4/ISI/887 Data path integrity	-	-	-	X	X	X	X	-	X	-	X
HYDR4	I4C/ISI/887 Data path integrity	-	-	-	-	-	-	X	X	X	X	X
ISTU	639/9639 Data path integrity	X	X	-	-	-	-	-	-	-	-	-
MYP	7152 PROM test	X	X	X	X	X	X	X	-	X	-	X
MY8	66X/844 Memory subsystem test	X	X	X	X	X	X	X	-	X	-	X
MY9	66X/844 Memory diagnostic	X	X	X	X	X	X	X	-	X	-	X
MTC	7X2X Subsystem diagnostic	X	X	X	X	X	X	X	-	X	-	X
UESM	CYBER 180 ESM utility/monitor	X	X	X	X	X	X	X	-	X	-	X
UESM4	CYBER 180 ESM utility/monitor	-	-	-	-	-	-	X	X	X	X	X
UHYD	887 Monitor utility	-	-	-	X	X	X	X	-	X	-	X
UHYD4	887 Monitor utility	-	-	-	-	-	-	X	X	X	X	X

DEMOT AND DEMOT DIAGNOSTICS		MSL 151 CIP 810/815	MSL 151 CIP 825/830	MSL 152 CIP 835	MSL 153 CIP 840	MSL 153 CIP 845/850	MSL 153 CIP 855/860/870	CYBER 96X CIP 960	CYBER 96X CIP 962	MSL 155 CIP 990/995	MSL 155 CIP 992	MSL 155 CIP 994
MCX	DEMOT PP executive W/CM	X	X	X	X	X	X	X	X	X	X	X
MPX	DEMOT PP executeive W/O CM	X	X	X	X	X	X	X	X	X	X	X
MLC	DEMOT compiler	X	X	X	X	X	X	X	X	X	X	X
MLD	DEMOT maintenance driver	X	X	X	X	X	X	X	X	X	X	X
ATC	7021-3X Controller test	X	X	X	X	X	X	X	X	X	X	X
BSU	ESM tester/loader/monitor	X	X	X	X	X	X	X	X	X	X	X
CCM	CCC memory test	X	X	X	X	X	X	X	X	X	X	X
CLM	CCC loader & monitor	X	X	X	X	X	X	X	X	X	X	X
CPE	415 Card reader exerciser	X	X	X	X	X	X	X	-	X	-	X
CRE	405 Card punch exerciser	X	X	X	X	X	X	X	-	X	-	X
CRP	CYBERPLUS test	X	X	X	X	X	X	X	-	X	-	X
DFU	895/834/836 Disk format utility	X	X	X	X	X	X	X	X	X	X	X
DL8	7155/855 Large sector test	X	X	X	X	X	X	X	X	X	X	X
DMA	CYBERPLUS DMA test	X	X	X	X	X	X	X	-	X	-	X
DM8	895 Inline test monitor	-	-	-	X	X	X	X	X	X	X	X
DTC	7155/844 Test	X	X	X	X	X	X	X	X	X	X	X
DTI	ISD/895/DMA Data path integrity	X	X	X	X	X	X	X	X	X	X	X
D44	7054/844 Disk test	X	X	X	X	X	X	X	X	X	X	X
D88	7155/885 Disk test	X	X	X	X	X	X	X	-	X	-	X
FHC	7155-401 Controller test	X	X	X	X	X	X	X	-	X	-	X
FFU	FSC format utility	X	X	X	X	X	X	X	-	X	-	X
FLD	885 Large sector test	X	X	X	X	X	X	X	X	X	X	X
FLM	FSC test loader monitor	-	-	X	X	X	X	X	-	X	-	X
FMC	7155-x Controller test	X	X	X	X	X	X	X	X	X	X	X

(continued)

DEMOT AND DEMOT DIAGNOSTICS *(continued)*		MSL 151 CIP 810/815	MSL 151 CIP 825/830	MSL 152 CIP 835	MSL 153 CIP 840	MSL 153 CIP 845/850	MSL 153 CIP 855/860/870	CYBER 96X CIP 960	CYBER 96X CIP 962	MSL 155 CIP 990/995	MSL 155 CIP 992	MSL 155 CIP 994
FMD	885 Serial drive test	X	X	X	X	X	X	X	-	X	-	X
FMU	885 Format utility	X	X	X	X	X	X	X	X	X	X	X
FSD	834/836 subsystem test	X	X	-	-	-	-	-	-	-	-	-
FSM	FSC memory test	X	X	X	X	X	X	X	-	X	-	X
FTP	580 Printer test	X	X	X	X	X	X	X	-	X	-	X
F44	FSC disk subsystem test	X	X	X	X	X	X	X	-	X	-	X
F7X	FSC tape subsystem test	X	X	X	X	X	X	X	-	X	-	X
F88	FSC disk subsystem test	X	X	X	X	X	X	X	-	X	-	X
IST	639 Tape subsystem test	X	X	-	-	-	-	-	-	-	-	-
ITU	639 Tape utility	X	X	-	-	-	-	-	-	-	-	-
LCI	LCN 170 device interface	X	X	-	X	X	X	X	X	X	X	X
LCM	LCN memory test	X	X	X	X	X	X	X	X	X	X	X
LCN	LCN confidence test	X	X	X	X	X	X	X	X	X	X	X
LPE	512 Printer exerciser test	X	X	X	X	X	X	X	-	X	-	X
LP1	512 Line printer test	X	X	X	X	X	X	X	-	X	-	X
LSP	ESM/STORNET low speed port	X	X	X	X	X	X	X	X	X	X	X
MSD	STORNET side door port	X	X	X	X	X	X	X	X	X	X	X
MSM	STORNET subsystem memory	X	X	X	X	X	X	X	X	X	X	X
NDM	LCN NAD memory test	X	X	X	X	X	X	X	X	X	X	X
NDP	LCN NAD instruction test	X	X	X	X	X	X	X	X	X	X	X
NDT	LCN NAD trunk test	X	X	X	X	X	X	X	X	X	X	X
NIP	5870 Non-impact printer	X	X	X	X	X	X	X	-	X	-	X
NLM	LCN NAD loader/monitor	X	X	X	X	X	X	X	X	X	X	X
PDP	PDP interface test	X	X	X	X	X	X	X	X	X	X	X

(continued)

DEMOT AND DEMOT DIAGNOSTICS *(continued)*		MSL 151 CIP 810/815	MSL 151 CIP 825/830	MSL 152 CIP 835	MSL 153 CIP 840	MSL 153 CIP 845/850	MSL 153 CIP 855/860/870	CYBER 96X CIP 960	CYBER 96X CIP 962	MSL 155 CIP 990/995	MSL 155 CIP 992	MSL 155 CIP 994
RT5	6671 Communications test	X	X	X	X	X	X	X	-	X	-	X
TT3	6675 Communications test	X	X	X	X	X	X	X	-	X	-	X
TFE	2550 Emulator communications test	X	X	X	X	X	X	X	-	X	-	X
TFF	2550 Coupler diagnostic	X	X	X	X	X	X	X	-	X	-	X
TFL	2550 Offline diagnostic loader	X	X	X	X	X	X	X	-	X	-	X
T6X	7021-2X/66X Tape test	X	X	X	X	X	X	X	-	X	-	X
T7X	7021-3X/67X Tape test	X	X	X	X	X	X	X	X	X	X	X
ULD	2550 Emulator upline dump	X	X	X	X	X	X	X	-	X	-	X

MSL Installation Procedures E

This appendix outlines the sequences of steps for installing CIP components, including MSL 15X, in shared-disk and maintenance only modes. The procedures guide you step-by-step through the general installation process, but refer you to other sections of this manual for detailed instructions about the utilities used. You are assumed to be familiar with components used as they are also described in other sections of this manual.

MSL 15X Installation Procedure

Use one of the following procedures to install CIP tape components on a Model 810, 815, 825, 830, 835, 840, 845, 850, 855, 860, 990, CYBER 960, or CYBER 994 Computer System prior to installing the operating system in a shared-disk environment.

Preparatory Procedures

1. Gather installation materials.
2. Read documentation.
3. Contact site analyst to coordinate MSL and operating system installation activities and prepare an installation plan.

Gather Installation Materials

___ 1. CYBER Initialization Package (CIP).

Locate the CIP for your computer system. This kit contains a software release bulletin (SRB), a CIP binary tape, and documentation for your system. Be sure you have the correct kit as defined by the FCA for your system.

The tape provided in a CIP field release kit includes common test initialization (CTI), Hardware Initialization and Verification Software (HIVS)/Maintenance Software Library (MSL), environment interface (EI), microcode, command buffers, and most model-dependent tests.

___ 2. Disk on which to install CIP components and the operating system.

When performing an initial install, ensure that this disk does not contain any permanent files that must be preserved.

Read Documentation

___ 1. Read the SRB furnished with CIP.

___ 2. Review the Field Change Announcement (FCA). Check compatibility of the computer system with microcode, MSL, and operating system tapes provided.

___ 3. Review the following documentation:

- Coldstart and warmstart procedures in the appropriate CIP reference manual.
- CTI/MSL Disk Area Utility (CAU) in the appropriate CIP reference manual (review options A, B, C, and D).
- Tape-to-Disk Utility (TDX) in section 6 of this manual (review options A, B, C, D, and F).
- Command buffer maintenance in section 4 of this manual.

___ 4. Review the displays and options described in the appropriate CIP reference manual.

Contact Site System Analyst

When installing or updating CIP and the operating system on a shared-disk, it is imperative that you and the site system analyst schedule computer time and perform the installation as a team to ensure that no system degradation occurs.

In shared-disk mode, a maximum of 20 megabytes of disk storage are allocated for MSL. If you want to enlarge the MSL area, you must first obtain customer permission to do so, then install in maintenance only mode.

Initial Auto-Install Procedure (Shared-Disk, Models 810, 815, 825, and 830)

Use the following procedure to initially install components of an unmodified CIP tape to the CIP device in a shared-disk environment for models 810, 815, 825, or 830. This procedure installs either a predefined subset of MSL tests or all of MSL onto the CIP device, depending upon whether a Short or Full Installation option is selected. The Short option should normally be selected. If additional tests are required, CMSE can be used to load individual tests.

The initial auto-install procedure assumes that this is the initial installation of CIP. The procedure reserves 12 megabytes of disk storage when the Short Installation option is selected and 20 megabytes of disk storage when the Full Installation is selected. CIP installation requires dedicated machine time, and at least one tape drive and one disk unit.

The procedure assumes that controlware has been loaded into the peripheral controllers. If the controlware isn't loaded, refer to coldstart procedures in the appropriate CIP reference manual.

___ 1. Ensure that the tape (639 or 66X) and disk (834, 836, 844, or 885) controlware are present and functioning properly and perform a system warmstart.

  ___ a. Mount the CIP tape without the write-enable ring and ready the unit.

  ___ b. Enter the values for a warmstart from tape through the MAINTENANCE OPTIONS display. Refer to warmstart procedures in the appropriate CIP reference manual. After a successful warmstart, the INITIAL OPTIONS display should appear.

___ 2. Press **(CR)** to select the default option, Build Deadstart Disk.

CAUTION

The next step destroys all information currently on the deadstart disk/CIP device prior to installing CIP. Before proceeding, be sure you have a back-up copy of any information on the deadstart disk (CIP device) that you want to preserve.

___ 3. Enter one of the following characters:

S To initialize the CIP device and to install the majority of CIP, including a predefined set of the most frequently used diagnostics.

F To initialize the CIP device and to install all of the CIP.

___ 4. Enter the channel, equipment, and unit numbers of the CIP device as prompted.

___ 5. The message `INSTALLATION COMPLETE` appears upon completion of CIP installation. The system is now ready to call CMSE, install the operating system, or to execute CTI utilities.

Initial Auto-Install Procedure (Shared-Disk, Models 835, 840, 845, 850, 855, 860, 990, CYBER 960, and CYBER 994)

Use the following procedure to perform an initial installation of the components of an unmodified CIP tape to the CIP device in a shared-disk environment for models 835, 840, 845, 850, 855, 860, 990, CYBER 960, and CYBER 994. This procedure installs the entire MSL onto the CIP device.

The initial auto-install procedure assumes that this is the initial installation of CIP. The procedure reserves 20 megabytes of disk storage for the CIP and requires dedicated machine time. At least one tape drive and one disk unit must be available.

The procedure assumes that controlware has been loaded into the peripheral controllers. If the controlware isn't loaded, refer to coldstart procedures in the appropriate CIP reference manual.

___ 1. Ensure that the tape (66X) and disk (844, 885, or 895) controlware are present and functioning properly, and perform a system warmstart.

  ___ a. Mount the CIP tape without the write-enable ring and ready the unit.

  ___ b. Set the deadstart panel/program of a model 835, 840, 845, 850, 855, 860, 990, CYBER 960, or CYBER 994 for a warmstart from tape. Refer to Warmstart Procedures in the appropriate CIP reference manual. After a successful warmstart, the INITIAL OPTIONS display should appear.

___ 2. Press **(CR)** to select the default option, Build Deadstart Disk.

CAUTION

The next step destroys all information currently on the CIP device, prior to installing the CIP. Before proceeding, ensure that you have a back-up copy of any information on the CIP device that you wish to preserve.

___ 3. Type `I` to select the Initial Installation option. This option initializes the CIP device and installs CIP.

___ 4. Enter the channel, equipment, and unit numbers of the CIP device as prompted.

___ 5. The message `INSTALLATION COMPLETE` appears upon completion of CIP installation. The system is now ready to call CMSE, install the operating system, or execute CTI utilities.

Update Auto-Install Procedure (Shared-Disk)

Use the following procedure to update CIP components on the CIP device, which have been installed previously using the Initial Installation Procedure.

The update auto-install procedure installs the CIP tape components on the CIP device, while it preserves operating system information on the CIP device (including all permanent files). The update installation procedure requires dedicated machine time. At least one tape drive and one disk unit must be available.

The procedures assume that controlware has been loaded into the peripheral controllers. If the controlware isn't loaded, refer to coldstart procedures in section 2 of this manual and in the appropriate CIP reference manual.

___ 1. Ensure that the tape (639 or 66X) and disk (834, 836, 844, 885, or 895) controlware are present and functioning properly and perform a system warmstart.

   ___ a. Mount the CIP tape without the write-enable ring and ready the unit.

   ___ b. Set the deadstart panel/program of a model 835, 840, 845, 850, 855, 860, 990, CYBER 960, or CYBER 994 for a warmstart from tape; for a model 810, 815, 825, or 830, enter the values for a warmstart from tape through the MAINTENANCE OPTIONS display. Refer to warmstart procedures in the appropriate CIP reference manual. After a successful warmstart, the INITIAL OPTIONS display should appear.

___ 2. Press **(CR)** to select the default option, Build Deadstart Disk.

___ 3. Type `U` to select the Update option. This option installs the CIP and preserves other information previously stored on the disk.

___ 4. Enter the channel, equipment, and unit numbers of the CIP device as prompted.

___ 5. The message `INSTALLATION COMPLETE` appears upon completion of CIP installation. The system is now ready to call CMSE, install the operating system, or execute CTI utilities.

Manual Installation Procedure (Shared-Disk)

Use the manual installation procedure when you have modified the components of the CIP tape or when you need tests other than the predefined subset of tests.

Install CIP Components

Perform the following sequence to install CIP components in a shared-disk mode.

___ 1. Ensure that the tape (639 or 66X) and disk (834, 836, 844, 885, or 895) controlware are present and functioning properly, and perform a system warmstart.

  ___ a. Mount CIP tape without the write-enable ring and ready the unit.

  ___ b. Set the deadstart panel/program of a model 835, 840, 845, 850, 855, 860, 990, CYBER 960, or CYBER 994 for a warmstart from tape; for a model 810, 815, 825, or 830, enter the values for a warmstart from tape through the MAINTENANCE OPTIONS display. Refer to warmstart procedures in the appropriate CIP reference manual. After a successful warmstart, the INITIAL OPTIONS display appears.

___ 2. Install CTI module on the CIP device.

  ___ a. Type B while displaying the INITIAL OPTIONS display. The BUILD DEADSTART DISK display appears.

  ___ b. Type M. The MANUAL OPERATIONS display appears.

  ___ c. Type C to install CTI. The C OPTION display appears.

  ___ d. Press **(CR)**; the system then requests channel, equipment, and unit numbers of the disk device. This serves to properly prepare a disk which has not previously had CTI, MSL, or HIVS installed on it. Reply to the channel, equipment and unit number enquiries.

  ___ e. Press **(CR)**. The following message appears:

```
ENTRY OF (CR) WILL CAUSE
RELEASE OF CTI-MSL/HIVS RESERVED
DISK SPACE
```

  ___ f. Press **(CR)**. The following message appears if the operation is successful:

```
RELEASE COMPLETE
(CR) TO PROCESS DIFFERENT DEVICE
```

  ___ g. Press **(CR)**. The C OPTION display reappears.

  ___ h. Press **(CR)**. The following warning message appears.

```
PERMANENT FILES MAY BE LOST IF CTI IS NOT ALREADY INSTALLED ON THIS
 DEVICE

(CR) TO CONTINUE
```

___ i. Press **(CR)**. The system now requests channel, equipment, and unit numbers of disk device. Enter channel, equipment, and unit number for device.

The following message appears when CTI is loaded successfully:

```
INSTALL COMPLETE
(CR) TO PROCESS DIFFERENT DEVICE
```

___ j. If your site has more than one system disk, press **(CR)** and repeat steps 2d through 2i for each disk.

___ k. Execute the deadstart; the INITIAL OPTIONS display appears.[1] The version of CIP is indicated at the bottom of the display.

___ 3. Install CTI/MSL Disk Area (CDA) Utility.

___ a. Type B while displaying the INITIAL OPTIONS display. The BUILD DEADSTART DISK display appears.

___ b. Type M. The MANUAL OPERATIONS display appears.

___ c. Select the D option. Then enter the disk and tape channel and unit number as prompted by the display. See CAU utility in the appropriate CIP reference manual.

If the disk unit selected for the CDA utility is reserved by another controller, the following message appears:

```
DISK UNIT RESERVED
```

Clear the reserved status of the disk unit to initiate an automatic retry.

If the disk selected for the CDA utility is a fixed module drive whose READ ONLY switch is set, the following message appears:

```
READ ONLY SELECTED.
```

Turn off the READ ONLY switch and press **(CR)** to initiate an automatic retry.

After the switch is cleared, the following message appears:

```
NO SPACE RESERVED FOR CTI/MSL
DISK AREA.  MSL/HVS OR OS FILES
MAY BE LOST IF THE OPERATION
CONTINUES.  ENTER (CR)
TO CONTINUE OR DEADSTART.
```

___ d. Press **(CR)** to use the currently selected disk. When the CDA is successfully installed, the CAU INITIAL OPTIONS display appears.

___ e. If your site has more than one system disk, press **Backspace** and repeat steps 3d and 3e for each disk.

1. Refer to the appropriate CIP reference manual for model-specific/console-specific deadstart procedures.

___ 4. Install microcode to the CIP device.

  ___ a. Type B while displaying the CAU INITIAL OPTIONS display.

  ___ b. When installation is complete, the CAU INITIAL OPTIONS display appears.

___ 5. Install default parameter deck to the CIP device from UDS tape.

  ___ a. Type A while displaying the CAU INITIAL OPTIONS display.

  ___ b. When installation is complete, the CAU INITIAL OPTIONS display appears.

___ 6. Install EI to CTI/MSL disk area.

  ___ a. Type C while displaying the CAU INITIAL OPTIONS display.

  ___ b. When installation is complete, the CAU INITIAL OPTIONS display appears.

___ 7. Install CC634B system console driver (SCD) to the disk.

  ___ a. Type D while displaying the CAU INITIAL OPTIONS display.

  ___ b. The CAU INITIAL OPTIONS display reappears once the installation is complete.

___ 8. Install monitor display driver (MDD) to disk.

  ___ a. Type C while displaying the CAU INITIAL OPTIONS display.

  ___ b. The CAU INITIAL OPTIONS display reappears once the installation is complete.

  ___ c. Perform the steps neccessary to return to the INITIAL OPTIONS display.

___ 9. Install MSL module to disk.

  ___ a. Type B while displaying the INITIAL OPTIONS display. The BUILD DEADSTART DISK display appears.

  ___ b. Type M. The MANUAL OPERATIONS display appears.

  ___ c. Type T. The console displays:

```
              TDX
DISK AND TAPE TRANSFER UTILITY
        CR TO CONTINUE
```

  ___ d. Press (CR), followed by TDX parameters as described in Section 6 of this manual. The TDX option display appears upon completion of these entries.

  ___ e. Type A to build MSL from tape.

___ f. Type F to select MSL/OS Shared Disk mode. Programs are installed at a predefined area of the disk.

The following message is displayed:

```
SAVE COMMAND BUFFER AREA
Y = YES   N = NO
```

___ g. Type N in response to the above message (no CR is required). TDX initializes the program name table (PNT) and sector reservation table (SRT) and presents the following display:

```
COPY FROM
-CR- = 1ST NAME
```

___ h. Press **(CR)** to cause TDX to begin copying at the first program it encounters. When the COPY FROM selection is complete, TDX presents the following display:

```
COPY THRU
-CR- = LAST NAME
```

___ i. Press **(CR)** to instruct TDX to copy to the last program on the tape. For tape-to-disk copies, TDX has the ability to verify data written to disk. TDX presents the following display:

```
DATA VERIFY (Y/N)
```

___ j. Type Y. TDX transfers each program to disk, displaying the name of each program as it is copied to the disk. TDX skips over any command buffers located on the tape. Upon completion of the copy operation, TDX displays the last cylinder, track and sector used for the copy. Press the **Space Bar** to display the last available cylinder for the complete MSL build.

NOTE

If the SRT FULL message appears, the edited MSL is too large for the predefined disk area. Therefore, you must either use more disk space and install in maintenance only mode, or use an alternate tape editing method and install a partial MSL. In either case, you must deadstart before you can proceed.

___ k. Press the **Space Bar** to clear the display and to display a reduced set of TDX options.

___ 10. Install command buffers to disk.

___ a. Type B when the TDX options display is present. TDX presents the following display:

```
COPY FROM
-CR- = 1ST NAME
```

___ b. Press **(CR)** to prompt TDX to begin copying with the first command buffer it encounters. TDX then presents the following display:

```
COPY THRU
-CR- = LAST NAME
```

___ c. Press **(CR)** to instruct TDX to copy to the last command buffer on the tape. For tape-to-disk copies, TDX has the ability to verify data written to disk. TDX presents the following display:

```
DATA VERIFY (Y/N)
```

___ d. Type Y. TDX transfers each command buffer to the disk, displaying the name of each command buffer as it is copied to the disk. Upon completion of the copy operation, TDX displays the last cylinder, track, and sector used for the copy.

___ e. Press the **Space Bar** to clear the message and to display a reduced set of TDX options.

___ 11. The system is now ready to install the operating system. The site analyst should install the operating system according to the installation plan agreed upon previously.

MSL 15X Installation Procedure (Maintenance Only)

Use this procedure to install an unedited CIP binary tape and microcode during installation and checkout of a model 810, 815, 825, 830, 835, 840, 845, 850, 855, 860, 990, or a CYBER 960 or 994 computer system. You should also use this procedure if you have permission from the customer to install an edited MSL tape which exceeds the disk limitations for shared-disk mode.

Preparatory Procedures

- Gather installation materials.
- Read documentation.

Gather Installation Materials

Perform the following preparatory steps before proceeding with the MSL maintenance installation procedures.

___ 1. Locate the CIP for your computer system. This kit contains a SRB, a CIP binary tape, and documentation for your system. Be sure you have the correct kit as defined by the FCA for your system.

The tape provided with the system hardware includes CTI, MSL, EI, microcode, command buffers, and model-dependent tests.

___ 2. Appoint a scratch disk on which MSL will be installed.

___ 3. Locate a scratch tape on which you will save modified command buffers.

Read Documentation

___ 1. Read the SRB furnished in the CIP.

___ 2. Review the FCA. Check compatibility of the computer system with CIP and operating system tapes provided.

___ 3. Review the following documentation.

- Coldstart, warmstart, and power-on initialization procedures in the appropriate CIP reference manual.
- CTI/MSL Disk Area Utility (CAU) in the appropriate CIP reference manual (review options C and D).
- Tape-to-Disk Utility (TDX) in section 6 (options A, B, C, D, and F).
- Command Buffer Modification in section 4 of this manual.

___ 4. Review the displays and options described in the appropriate CIP reference manual.

Installation Procedure (Maintenance Only)

This procedure assumes that this is the initial installation of CIP components and that a disk unit is dedicated for maintenance (maintenance only mode). For MSL 15X shared-disk situations, use the MSL 15X Shared-Disk Installation Procedures provided earlier in this appendix.

___ 1. Ensure that the tape (639 or 66X) and disk (834, 836, 844, 885, or 895) controlware are loaded and functioning properly.

___ a. Mount the CIP tape without the write enable ring. Ready the tape unit.

___ b. Refer to Coldstart Procedures in the appropriate CIP reference manual; perform the appropriate coldstart. A coldstart loads the tape/disk controller with controlware.

___ c. Set the deadstart panel/program of a model 810, 815, 825, 830, 835, 840, 845, 850, 855, 860, 990 or a CYBER 960 or 994 for a warmstart from tape. Refer to the appropriate CIP reference manual for specific warmstart procedures.

___ 2. Install CTI module on a disk.

___ a. Type B while displaying the INITIAL OPTIONS display. The BUILD DEADSTART DISK display appears.

___ b. Type M. The MANUAL OPERATIONS display appears.

___ c. Type C to install CTI. The C option display appears.

___ d. Type R. The system requests channel, equipment, and unit numbers for the disk device. This serves to properly prepare a disk which has not previously had CTI, MSL, or HIVS installed on it. Enter channel, equipment and unit number for device.

___ e. Press **(CR)**. The following message appears:

```
ENTRY OF (CR) WILL CAUSE
RELEASE OF CTI-MSL/HIVS RESERVED
DISK SPACE
```

___ f. Press **(CR)**. The following message appears if the operation is successful:

```
RELEASE COMPLETE
(CR) TO PROCESS DIFFERENT DEVICE
```

___ g. Press **(CR)**. The C option display reappears.

___ h. Press **(CR)**. The following warning message appears.

```
           *WARNING*
PERMANENT FILES MAY BE LOST
IF CTI IS NOT ALREADY
INSTALLED ON THIS DEVICE

(CR) TO CONTINUE
```

___ i. Press **(CR)**. The system requests channel, equipment, and unit numbers of the disk device. Reply with the correct channel, equipment, and unit numbers.

The following message appears if CTI is loaded successfully:

```
INSTALL COMPLETE
(CR) TO PROCESS DIFFERENT DEVICE
```

___ j. Execute the deadstart; the INITIAL OPTIONS display appears.[2]

___ 3. Install microcode to CTI/MSL disk area.

___ a. Type B while displaying the INITIAL OPTIONS display. The BUILD DEADSTART DISK display appears.

___ b. Type M. The MANUAL OPERATIONS display appears.

___ c. Type D. Reply to the the disk channel and unit number enquiries as prompted by the display. See the CAU utility in the appropriate CIP reference manual for details.

If the disk unit selected is reserved by another controller, the following message appears:

```
DISK UNIT RESERVED
```

Clear the reserved status of the disk unit to initiate an automatic retry.

If the disk selected is a fixed module drive whose READ ONLY switch is set, the following message appears:

```
READ ONLY SELECTED.
```

Turn off the READ ONLY switch and press **(CR)** to initiate automatic retry.

---

2. Refer to the appropriate CIP reference manual for model-specific/console-specific deadstart procedures.

After the switch is cleared, the following message appears:

```
NO SPACE RESERVED FOR CTI/MSL
DISK AREA.  MSL/HVS OR OS FILES
MAY BE LOST IF THE OPERATION
CONTINUES.  ENTER (CR)
TO CONTINUE OR DEADSTART
```

___ d. Press **(CR)** to use the currently selected disk. When the CDA is successfully installed, the CAU INITIAL OPTIONS display appears.

___ e. Type `C`. When prompted, enter the tape type, channel, equipment, and unit numbers, or press **(CR)** at each prompt to accept the default value. Enter the microcode type for your particular model of computer system. Refer to the CAU utility in the appropriate CIP reference manual for detailed instructions.

___ f. When installation is complete, the CAU INITIAL OPTIONS display appears.

___ 4. Install environment interface (EI) to the CTI/MSL disk area.

___ a. Type `D` while displaying the CAU INITIAL OPTIONS display.

___ b. When installation is complete, the CAU INITIAL OPTIONS display appears.

___ c. Execute the deadstart; the INITIAL OPTIONS display appears.[3]

___ 5. Install MSL module to disk.

___ a. Type `B` while displaying the INITIAL OPTIONS display. The BUILD DEADSTART DISK display appears.

___ b. Type `M`. The MANUAL OPERATIONS display appears.

___ c. Type `T`. The console displays:

```
            TDX
DISK AND TAPE TRANSFER UTILITY
      CR TO CONTINUE
```

___ d. Press **(CR)**, then enter TDX parameters as described in section 6 of this manual. The TDX option display appears upon completion of these entries.

___ e. Type `A` to build MSL from tape.

___ f. Type `M` to select the Maintenance Only option when the initial display presented by option A appears.

___ g. Enter the starting cylinder, followed by a CR. Refer to the table provided in the DSRB for starting cylinder and disk type. If the disk is a scratch disk, you can use cylinder number 0. However, if the disk is to be used to hold the operating system after system checkout, use the values in the DSRB.

3. Refer to the appropriate CIP reference manual for model-specific/console-specific deadstart procedures.

The following message is displayed:

```
SAVE COMMAND BUFFER AREA
Y = YES   N = NO
```

___ h. Type N in response to the above message (No CR is required). TDX initializes the PNT and SRT and then presents the following display:

```
COPY FROM
-CR- = 1ST NAME
```

___ i. Press **(CR)** to prompt TDX to begin copying with the first program it encounters. When the COPY FROM selection is complete, TDX presents the following display:

```
COPY THRU
-CR- = LAST NAME
```

___ j. Press **(CR)** to instruct TDX to copy to the last program on the tape. For tape-to-disk copies, TDX has the ability to verify data written to disk. TDX presents the following display:

```
DATA VERIFY (Y/N)
```

___ k. Type Y. The name of each program is displayed as it is transferred to disk: TDX skips over any command buffers located on the tape. Upon completion of the copy operation, TDX displays the last cylinder, track and sector used for the copy. Press the **Space Bar** to display the last available cylinder for the complete MSL build.

___ l. Press the **Space Bar** to clear the message and display a reduced set of TDX options.

___ 6. Install command buffers to disk.

___ a. Type B while the TDX options display is presented. TDX presents the following display:

```
COPY FROM
-CR- = 1ST NAME
```

___ b. Press **(CR)** to prompt TDX to begin copying with the first command buffer it encounters. TDX then presents the following display:

```
COPY THRU
-CR- = LAST NAME
```

___ c. Press **(CR)** to instruct TDX to copy through the last command buffer on the tape. For tape-to-disk copies, TDX has the ability to verify data written to disk. TDX then presents the following display:

```
DATA VERIFY (Y/N)
```

___ d. Type Y. The name of each command buffer is displayed as it is copied to disk. Upon completion of the copy operation, TDX displays the last cylinder, track and sector used for the copy.

___ e. Execute the deadstart; the INITIAL OPTIONS display appears.[4]

___ 7. Deadstart to CIP device.

___ a. Refer to the appropriate CIP reference manual for procedures on how to bring up the initial CMSE display shown in section 2 of this manual.

___ 8. Modify command buffers.

___ a. While displaying the Initial CMSE display, enter parameters to define the system configuration. Follow each entry with a CR.

___ b. Press **(CR)** after all system configuration changes have been made. The CMSE A and B display headers appear. Refer to Displays in section 3 of this manual.

___ c. Display and modify the command buffers as directed by comments embedded in each command buffer. Refer to Command Buffer Modification in section 4 of this manual for detailed instructions.

___ d. Execute the deadstart; the INITIAL OPTIONS display appears.[4]

___ 9. Copy command buffers to tape.

___ a. Mount a scratch tape with a write ring and ready the tape unit.

___ b. Type B character while displaying the INITIAL OPTIONS display. The Build Deadstart Disk display appears.

___ c. Type M. The MANUAL OPERATIONS display appears.

___ d. Type T while displaying the MANUAL OPERATIONS display.

___ e. Type F when the TDX options display is presented. TDX then presents the following display:

```
COPY FROM
-CR- = 1ST NAME
```

___ f. Press only **(CR)** to prompt TDX to begin copying the first command buffer listed in the command buffer PNT, or enter the name of the first command buffer to be copied to tape and follow it with a CR. TDX then presents the following display:

```
COPY THRU
-CR- = LAST NAME
```

___ g. Either press **(CR)**, which instructs TDX to copy to the last name in the PNT, or enter the name of the last command buffer to be copied and follow it with a CR. TDX then presents the following message:

```
IS DATA ON THE TAPE VALID (Y/N)
```

___ h. Type Y to add command buffers to an existing tape, or an N to create a new tape of command buffers. Following either response, TDX executes the operation and displays a reduced set of copy options.

---

4. Refer to the appropriate CIP reference manual for model-specific/console-specific deadstart procedures.

___ i. Execute the deadstart; the INITIAL OPTIONS display appears.[5]

___ 10. The system is now ready to perform hardware checkout. When hardware checkout is complete, contact the site analyst and jointly prepare a plan to install the operating system.

5. Refer to the appropriate CIP reference manual for model-specific/console-specific deadstart procedures.

Command Summary

Peripheral Processor Commands	Description	Page Number
AP,num,fwa	Display PP Memory on A Display	3-29
AU,num,fwa	Select PP Dump on A Display	3-32
BP,num,fwa	Display PP Memory on B Display	3-29
BU,num,fwa	Select PP Dump on B Display	3-32
CP,num,name,fwa,p0-p17	Load PP Program from MSL Device	3-36
DI,z,fwa	Change PP Display Block	3-32
DP,num	Deadstart PP	3-39
EH,num,adrs,data	Enter PP Memory Hexadecimal Data	3-34
EP,num,adrs,data	Enter PP Memory	3-34
HT,num	Halt PP Program Execution	3-40
KP,num,fwa,lwa+1,data,incr	Clear PP Memory	3-35
LTxx,name,fwa,,p0-p17	Call Test into PP and Use Library Overlays	3-37
MP,spp,dpp	Move PP Memory	3-35
PP,num,fwa	Display Hexadecimal PP Memory	3-29
RP,num,p1-p10	Repetitive Deadstart	3-40
RU,num,adrs	Execute PP Program	3-39
TLxx,name,fwa,,p0-p17	Call Test into PP and Use CM Overlays	3-38
*WP,base,num,name,fwa, lwa+1	Write PP Program to MSL Disk	3-41

Central Memory Commands	Description	Page Number
AB,fba	Display Hexadecimal CM on A Display, Byte Address	3-44
AC,fwa	Display Octal CM on A Display	3-42
AD,fwa	Display Octal CM on A Display	3-42
AH,fwa	Display Hexadecimal CM on A Display	3-47
AV,pva,sta,pta,psm,ptl	Display Virtual Memory on A Display	3-49
AX,fwa	Display CM in Exchange Package Format on A Display	3-50
BB,fba	Display Hexadecimal CM on B Display, Byte Address	3-44
BC,fwa	Display Octal CM on B Display	3-42
BD,fwa	Display Octal CM on B Display	3-42
BH,fwa	Display Hexadecimal CM on B Display	3-47
BV,pva,sta,pta,psm,ptl	Display Virtual Memory on B Display	3-49
BX,fwa	Display CM in Exchange Package Format on B Display	3-50
CC,name,fwa	Load CM Program from MSL Device	3-55
CF,Pnum	Stop Monitoring CM for CMSE Calls	3-56
CH,adrs1,adrs2,wdcnt	Compare Hexadecimal Data in CM	3-57
CN,Pnum,base,adrs	Monitor CM for CMSE Calls	3-56
EB,badrs,data	Enter Byte of CM Data	3-52
EC,base,adrs,data	Enter Word of CM Data	3-52
KC,base,fwa,lwa+1,data,incr	Set/Clear Block of CM	3-53
MC,base,sadr,dadr,nmbr	Move Central Memory	3-54
*WC,base,name,fwa,lwa+1	Write CM Program to MSL Disk	3-57

Central Memory Commands	Description	Page Number
XAf,d	Set A Reg Position at CM Address f to d	3-54
XBf,d	Set B Reg Position at CM Address f to d	3-54
XPf,d	Set P Reg Position at CM Address f to d	3-55

Central Processor Commands	Description	Page Number
CU,Pnum	Assign the Job Display to CPU num	3-58
PD,Pnum	Assign default CPU	3-58

Control Store Commands	Description	Page Number
AK,Pnum,L,fwa	Select A Display for CS Display	3-61
AK,Pnum,My,fwa	Select A Display for CS Display	3-59
AS,Pnum,mem,adrs	Select Soft Control Memory Display	3-70
BK,Pnum,L,fwa	Select B Display for CS Display	3-61
BK,Pnum,My,fwa	Select B Display for CS Display	3-59
BS,Pnum,mem,adrs	Select Soft Control Memory Display	3-70
CK,Pnum,My,name,fwa	Load CS Program from MSL Device	3-65
CS,Pnum,name,mem,adrs	Load Soft Control Memory from MSL	3-83
CS,Pnum,name,t,fwa	Load Soft Control Memory from MSL	3-83
DK,Pnum,My,adrs	Deadstart Control Store	3-67
EK,Pnum,L,adrs,byteno,data	Enter CS	3-64
EK,Pnum,My,adrs,byteno, data	Enter CS	3-64
ES,Pnum,mem,adrs,byte,data	Alter Soft Control Memory	3-85
ES,Pnum,t,adrs,data	Alter Soft Control Memory	3-84
HK,Pnum	Halt Microcode Execution	3-67
RK,Pnum,My,adrs,delay,C	Repetitive Deadstart Control Store	3-68
VK,Pnum,My,name,fwa	Load CS Program from MSL Device and Verify	3-66
*WK,Pnum,My,name,fwa, 1wa+1	Write CS to MSL Disk	3-69
*WS,Pnum,name,mem,fwa, 1wa+1	Write Soft Control Memory to MSL Disk	3-86

Special Memory Commands	Description	Page Number
AR,Pnum,mem	Select Special Memory Display	3-87
AR,Pnum,SCR,name,reg	Select Special Memory Display	3-88
AT,Pnum,disp,adrs	Control Word Display	3-90
BR,Pnum,SCR,name,reg	Select Special Memory Display	3-88
BR,Pnum,mem	Select Special Memory Display	3-87
BT,Pnum,disp,adrs	Control Word Display	3-90
CT,Pnum,name,adrs	Load Control Word Program from MSL	3-93
ET,Pnum,L,adrs,byte,data	Enter Control Word	3-92
MX,adrs	Move Capture Buffer to CM for Display	3-92
VT,Pnum,name,adrs	Load Control Word Program from MSL	3-93
*WT,Pnum,name,fwa,1watl	Write Control Word to MSL Disk	3-93

Maintenance Channel Commands	Description	Page Number
AR,el,S	Select A Maintenance Register Display	3-94
AR,el,reg	Select A Maintenance Register Display	3-95
BR,el,S	Select B Maintenance Register Display	3-94
BR,el,reg	Select B Maintenance Register Display	3-95
CE,el	Clear Error Via Maintenance Channel	3-99
CX,el	Master Clear Via Maintenance Channel	3-99
ER,el,reg,data,mask	Enter Maintenance Register Data	3-98

Command Buffer Commands	Description	Page Number
(blank in first column) comment	Display Comment	3-106
AE,name,num	Display Command Buffers on A Display	3-100
AG,indx	Display Command Buffer Name Table on A Display	3-102
AM,linenum,msg	Display Message on A Screen	3-107
BE,name,num	Display Command Buffers on B Display	3-100
BG,indx	Display Command Buffer Name Table on B Display	3-102
BM,linenum,msg	Display Message on B Screen	3-107
*DB *	Delete All Command Buffers from Disk Library	3-108
*DB,name	Delete Command Buffer from Disk Library	3-108
DE,num,cmnd	Delete Command in Command Buffer	3-103
GO,name	Execute Command Buffer	3-104
IN,num,cmnd	Insert Command in Command Buffer	3-103
KE	Drop Command Buffer Editing Utility	3-106
RJ,name	Return Jump to Command Buffer	3-106
SP	Send Space Command	3-106
SQ,base,adrs,tshld,mem	Temporarily Halt Command Buffer Sequence	3-105
TB	Terminate Command Buffer	3-104
TB*	Terminate all Active and Waiting Command Buffers	3-104
TB,*	Terminate all Active and Waiting Command Buffers	3-104
*WB	Write Command Buffer to MSL Disk	3-108
.	Command Buffer Comment Line	3-107

CMSE Disk Library Commands	Description	Page Number
AF,indx	Display Program Name Table on A Display	3-109
AY,indx	Display Dayfile on A Display	3-111
BF,indx	Display Program Name Table on B Display	3-109
BY,indx	Display Dayfile on B Display	3-111
*DP,*DF	Delete Dayfile	3-109
*DP,name	Delete Program from MSL Disk	3-109
ED,msg	Enter Dayfile Command	3-112

CMSE Utility Commands	Description	Page Number
AA	Display Active Requests on A Display	3-113
*ACn	Activate Channel	3-120
AI,x	Display Help Information on A Display	3-114
AN	Null A Display	3-116
AS,Pnum,base,reg	Display Register File on A Display	3-115
BA	Display Active Requests on B Display	3-113
BI,x	Display Help Information on B Display	3-114
BN	Null B Display	3-116
BS,Pnum,base,reg	Display Register File on B Display	3-115
CM,Pnum,el,mgn	Clock Margins	3-118
CM,el,mgn	Clock Margins	3-118
CW,name,ch	Autoload Buffer Controller with Controlware from Library Device	3-118
*DCn	Deactivate Channel	3-120
DN,num	Do Not Monitor PP	3-117
*FCn,f	Send Function to Channel	3-121
*IAn	Input to Pseudo-A Register	3-121
LE,el,cond	Log Errors	3-120
LK	Lock CMSE Area of CM from Keyboard Commands	3-124
LR,ntp,ntp	Load Local/Remote Display Driver(s)	3-123
*OAn,x	Output Data Word on Channel	3-121
*OV,base,adrs	Load Monitor Overlays to CM	3-122
*RV	Revert to Monitor Overlays from Disk	3-124
UL	Unlock CMSE Area of CM from Keyboard Commands	3-124
UP,num	Monitor PP	3-117

Comments (continued from other side)

Please fold on dotted line;
seal edges with tape only.

FOLD

FOLD

FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
First-Class Mail Permit No. 8241 Minneapolis, MN

POSTAGE WILL BE PAID BY ADDRESSEE

CONTROL DATA
Technical Publications
ARH219
4201 N. Lexington Avenue
Arden Hills, MN 55126-9983

MSL 15X Reference Manual **60456530 W**

We would like your comments on this manual to help us improve it. Please take a few minutes to fill out this form.

Who are you?

- ☐ Manager
- ☐ Systems analyst or programmer
- ☐ Applications programmer
- ☐ Operator
- ☐ Other ______________________

How do you use this manual?

- ☐ As an overview
- ☐ To learn the product or system
- ☐ For comprehensive reference
- ☐ For quick look-up
- ☐ Other ______________________

What programming languages do you use? ______________________

How do you like this manual? Answer the questions that apply.

Yes	Somewhat	No	
☐	☐	☐	Does it tell you what you need to know about the topic?
☐	☐	☐	Is the technical information accurate?
☐	☐	☐	Is it easy to understand?
☐	☐	☐	Is the order of topics logical?
☐	☐	☐	Can you easily find what you want?
☐	☐	☐	Are there enough examples?
☐	☐	☐	Are the examples helpful? (☐ Too simple? ☐ Too complex?)
☐	☐	☐	Do the illustrations help you?
☐	☐	☐	Is the manual easy to read (print size, page layout, and so on)?
☐	☐	☐	Do you use this manual frequently?

Comments? If applicable, note page and paragraph. Use other side if needed.

Check here if you want a reply: ☐

Name ______________________ Company ______________________

Address ______________________ Date ______________________

Phone ______________________

Please send program listing and output if applicable to your comment.

CORPORATE HEADQUARTERS, P.O. BOX O, MINNEAPOLIS, MINN 55440
SALES OFFICES AND SERVICE CENTERS IN MAJOR CITIES THROUGHOUT THE WORLD

LITHO IN U.S.

CONTROL DATA