MME Interface Reference

(CRAY T90[™] Series)

HDM-008-A

Cray Research Proprietary

Cray Research, Inc.

Record of Revision

REVISION DESCRIPTION

August 1995. Original printing.

A March 1996. This revision corresponds to the MT-T2.2.0 offline diagnostic release.

Any shipment to a country outside of the United States requires a letter of assurance from Cray Research, Inc.

This document is the property of Cray Research, Inc. The use of this document is subject to specific license rights extended by Cray Research, Inc. to the owner or lessee of a Cray Research, Inc. computer system or other licensed party according to the terms and conditions of the license and for no other purpose.

Cray Research, Inc. Unpublished Proprietary Information — All Rights Reserved.

Autotasking, CF77, CRAY, CRAY-1, Cray Ada, CraySoft, CRAY Y-MP, CRInform, CRI/*Turbo*Kiva, HSX, LibSci, MPP Apprentice, SSD, SUPERCLUSTER, SUPERSERVER, UniChem, UNICOS, and X-MP EA are federally registered trademarks and Because no workstation is an island, CCI, CCMT, CF90, CFT, CFT2, CFT77, ConCurrent Maintenance Tools, COS, CRAY-2, Cray Animation Theater, CRAY APP, CRAY C90, CRAY C90D, Cray C++ Compiling System, CrayDoc, CRAY EL, CRAY J90, Cray NQS, Cray/REELlibrarian, CRAY S-MP, CRAY SUPERSERVER 6400, CRAY T3D, CRAY T3E, CRAY T90, CrayTutor, CRAY X-MP, CRAY XMS, CS6400, CSIM, CVT, Delivering the power . . ., DGauss, Docview, EMDS, GigaRing, HEXAR, IOS, ND Series Network Disk Array, Network Queuing Environment, Network Queuing Tools, OLNET, RQS, SEGLDR, SMARTE, SUPERLINK, System Maintenance and Remote Testing Environment, Trusted UNICOS, UNICOS MAX, and UNICOS/mk are trademarks of Cray Research, Inc.

Requests for copies of Cray Research, Inc. publications should be directed to:

CRAY RESEARCH, INC. Customer Service Logistics 1100 Lowater Road Chippewa Falls, WI 54729

Comments about this publication should be directed to:

CRAY RESEARCH, INC. Service Publications and Training 890 Industrial Blvd. Chippewa Falls, WI 54729

MME INTERFACE REFERENCE

Description of this Document	ix
ENVIRONMENT 0	1
Interface Components	1
Automatic or Manual Mode	1
Base Window Title	2
Version Number	2
Simulator or FEI Channel	2
Workstation Name or Channel Number	2
Copy Number	2
Menu Bar	2
Module and Test Selection Area	2
Total Pass and Error Counts	3
Control Buttons	3
Configuration Information	3
Status Information	4
Modes	4
Compose Mode	5
Base Window Title	5
Version Number	5
Simulator or FEI Channel	5
Workstation Name or Channel Number	6
Copy Number	6
Menu Bar	6
Sequence Editing Buttons	6
Total Pass and Error Counts	7
Control Buttons	7
Configuration Information	8
Status Information	8
Modes	9
Sequence Scroll Box	9

ENVIRONMENT 0 (continued)

Menu Button Commands	10
File -> Load -> Sequence	10
File -> Load -> Data	11
File -> Save -> Sequence	12
File -> Save -> Data	13
File -> Delete	14
File -> Print -> Root	15
File -> Print -> Screen	15
File -> Print -> Setup	15
File -> Dump	16
View -> Memory	18
Changing Memory	23
Using the Keyboard Accelerator	25
View -> Buffer	33
View -> Log	34
View -> Report	34
View -> Notes	35
Properties -> Environment -> ENV1	35
Properties -> Environment -> ENV2	35
Properties -> Partition	35
Properties -> Resource Allocation	36
Specifying Which CPU Writes and Reads Memory	36
Modifying the Spare-chip Table that MME Uses	37
Viewing the Current Working Directory and Changing the Concurrent Maintenance Check and Debug Messages	
Settings	39
Properties -> Enable Auto Apply	40
Properties -> Disable Auto Apply	40
Utilities -> Pattern	41
Utilities -> Find	43
Utilities -> Copy/Move	46
Utilities -> Configuration	47
Utilities -> Logic Monitor	47

ENVIRONMENT 0 (continued)

Reset -> Channel	47
Reset -> Server	47
Reset -> Configuration	47

ENVIRONMENTS 1 AND 2

51	

Interface Components	51
Base Window Title	52
Version Number	52
Simulator or FEI Channel	52
Workstation Name or Channel Number	52
Copy Number	52
Menu Bar	52
CPU Selection, Control Point, and Status Area	53
Controls	57
Configuration Information	58
Sections Scroll Box	59
Status Information	60
Control Points Scroll Box	60
Menu Button Commands	62
File -> Load -> Data	62
File -> Load -> Control Point	64
File -> Load -> Environment	68
File -> Load -> Layout	69
File -> Save -> Data	69
File -> Save -> Control Point	71
File -> Save -> Environment	73
File -> Save -> Layout	74
File -> Info	74
File -> Delete	75
File -> Print -> Root	76
File -> Print -> Screen	76
File -> Print -> Setup	77

ENVIRONMENTS 1 AND 2 (continued)

File -> Dump	78
View -> Memory	80
Changing Memory	86
Using the Keyboard Accelerator	88
View -> CPUs -> Control Point	97
View -> CPUs -> Pass Count (Global)	97
View -> CPUs -> Pass Count (Section)	98
View -> CPUs -> P Register	98
View -> CPUs -> CIP	98
View -> Register Dump	99
View -> Error Log	101
Memory Error Information	101
Register Parity Error Information	102
Shared Register Error Information	102
LAT Error Information	103
Unknown Error Information	103
View -> Memory Map	104
View -> Runtime Information -> Current	106
View -> Runtime Information -> Controller (Environment 2	100
Only)	108
View -> Listing -> Current	109
View -> Listing -> Controller	111
View -> Listing -> Other	111
View -> Notes	112
Edit -> Delete Control Point -> Selected	112
Edit -> Delete Control Point -> All	113
Edit -> Assign CPU(s)	113
Edit -> Deassign CPU(s)	113
Properties -> Environment -> ENV2 (Environment 1 Only)	113
Properties -> Environment -> ENV1 (Environment 2 Only)	113
Properties -> Environment -> ENV0	113
Properties -> Partition	114
Properties -> Resource Allocation	114

ENVIRONMENTS 1 AND 2 (continued)

114
117
119
120
121
123
105
127
130
132
132
133
133
135
139
142
143
143
143
144
144
144
145
145
146
146
147
147
148

ENVIRONMENTS 1 AND 2 (continued)

Reset -> I/O 1 -> Quadrant 3	148
Reset \rightarrow I/O 2 \rightarrow Quadrant 0	149
$Reset \rightarrow I/O \ 2 \rightarrow Quadrant \ 1 \dots \dots$	149
$Reset \rightarrow I/O \ 2 \rightarrow Quadrant \ 2 \ \ldots \ldots \ldots \ldots \ldots$	150
$Reset \rightarrow I/O \ 2 \rightarrow Quadrant \ 3 \dots \dots \dots \dots$	150
Reset \rightarrow I/O 3 \rightarrow Quadrant 0	151
$Reset \longrightarrow I/O \ 3 \longrightarrow Quadrant \ 1 \dots \dots$	151
$Reset \rightarrow I/O \ 3 \rightarrow Quadrant \ 2 \dots \dots \dots \dots$	152
$Reset \rightarrow I/O \ 3 \rightarrow Quadrant \ 3 \ \ldots \ldots \ldots \ldots \ldots$	152
Reset -> Shared 0	153
Reset -> Shared 1	153

Figures

Figure 1.	Environment 0 Interface Components (Automatic or Manual Mode)	1
Figure 2.	Environment 0 Interface Components (Compose Mode)	5
Figure 3.	Environment 0 Menu Quick Reference	49
Figure 4.	Environments 1 and 2 Interface Components	51
Figure 5.	Master CPU Indicator	54
Figure 6.	Example Interrupt Flag	54
Figure 7.	Example Controller Status Message	55
Figure 8.	Memory Map Example	105
Figure 9.	Mainframe Memory Allocation in Environment 2	115
Figure 10.	Environment 1 Menu Quick Reference	155
Figure 11.	Environment 2 Menu Quick Reference	157

Tables

Table 1.	Interrupt Flags	55
Table 2.	Controller Status Messages	56
Table 3.	Control Point Runtime Information Categories	107
Table 4.	Controller Runtime Information Categories	109
Table 5.	I/O Maintenance Modes	123
Table 6.	Environments 1 and 2 Debug Messages Settings	129

Description of this Document

This document is a reference to the user interface for the Mainframe Maintenance Environment (MME) application used to troubleshoot CRAY T90 series mainframes. This document describes the interfaces used with MME environments 0, 1, and 2. It also describes all available menu button commands. Figure 9, Figure 10, and Figure 11 illustrate all menu button commands available in environments 0, 1, and 2.

This document is one component of the MME documentation set, which also includes the following documents:

MME User Guide, publication number HDM-102-A.

This document describes how to use the MME environments for troubleshooting. It includes information about the internal functionality of MME.

MME Diagnostic Tests and Utilities, publication number HDM-103-A.

This document provides quick-reference information for all diagnostic tests and utilities you can use with MME.

MME Interface Reference

ENVIRONMENT 0

This section describes the interface components and menu button commands for MME environment 0.

Interface Components

The MME environment 0 interface, located in the base window, controls testing in environment 0. Environment 0 uses two distinct graphical user interfaces: one for automatic and manual modes and one for compose mode.

Automatic or Manual Mode

Figure 1 shows the MME environment 0 graphical user interface for automatic or manual mode. The paragraphs following the figure describe the components of the interface.

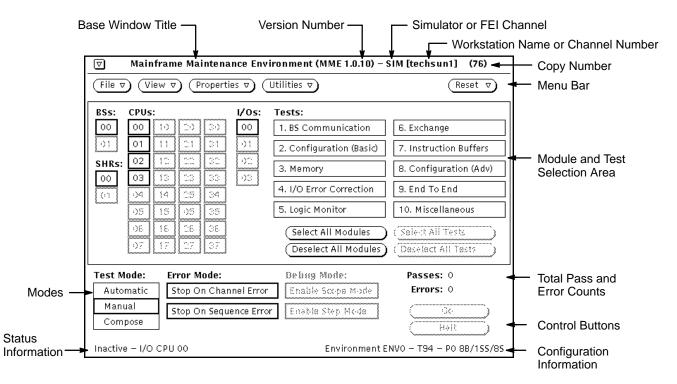


Figure 1. Environment 0 Interface Components (Automatic or Manual Mode)

Base Window Title

The base window title displays the name of the program: Mainframe Maintenance Environment.

Version Number

The version number indicates the version of MME you are using.

Simulator or FEI Channel

The simulator or front-end interface (FEI) channel indicator shows MME is running with the simulator (indicated by SIM) or an FEI channel (indicated by FEI CHN 0 for channel 0, FEI CHN 1 for channel 1, FEI CHN 2 for channel 2, etc.).

Workstation Name or Channel Number

The workstation name or channel number indicator lists the name of the workstation or the channel number on which MME is running.

Copy Number

The copy number indicates the copy of MME you are using. To set the copy number, start MME with the -copy option. If you start MME with the default copy number of 0, the MME base window does not display a copy number. For more information about starting MME with the -copy option, refer to the *MME User Guide*, publication number HDM-102-A.

Menu Bar

The menu bar contains five menu buttons: $(\underline{File v}), (\underline{View v}), (\underline{Properties v}), (\underline{Utilities v}), and (\underline{Reset v})$. For descriptions of the commands accessible from these menu buttons, refer to "Menu Button Commands" on page 10.

Module and Test Selection Area

In the module and test selection area, you can assign modules to the environment 0 tests. You can assign four types of modules to tests: BSs (boundary scan modules), SHRs (shared modules), CPUs (CPU modules), and I/Os (input/output modules).

Use the following settings to select environment 0 tests:

1.85 Communication	6. Eschange
2. Conlign ation (Bacid)	7. Instruction Buffers
3. Метьсу	8. Configuration (Adv)
4. NO En or Connection	9. End To End
5. Wait Monito	10. Miscellaneuus

Click on a setting to select a test. Use the <u>Select All Tests</u> and <u>Deselect All Tests</u> buttons to select or deselect all tests (automatic mode only). Use the <u>Select All Modules</u> and <u>Deselect All Modules</u> buttons to select or deselect all modules.

For more information about the environment 0 tests, refer to the *MME Diagnostic Tests and Utilities* document, publication number HDM-103-A.

Total Pass and Error Counts

The total pass count component (Passes) indicates the number of passes a test has completed. The total error count component (Errors) indicates the total number of errors found during the current test(s) executions. The pass and error counts are in decimal.

Control Buttons

Use the control buttons to start testing (\bigcirc Go) and stop testing (\bigcirc Halt).

Configuration Information

The configuration information component displays the current MME environment and configuration data (mainframe type; partition where MME is running; and number of sections, subsections, and banks). The System Configuration Environment (SCE) provides this information.

For example, the following configuration information indicates MME is in environment 0 (Environment ENV0); the mainframe type is a CRAY T94 system (T94); MME is running in partition 0 (P0); and the configuration consists of 8 banks (8B), 1 subsection (1SS), and 8 sections (8S).

Environment ENVO – T94 – PO 8B/1SS/8S

The configuration information component displays one of the following mainframe types:

Туре	Description
Tester	A tester with 1 CPU and 1 memory module
T94	A CRAY T94 mainframe
T916	A CRAY T916 mainframe
T932	A CRAY T932 mainframe

Status Information

The status information component displays the current state of the MME program, using the following messages:

Message	Description
Active	Test(s) are running.
Inactive	No test(s) are running.
I/O CPU ##	The specified CPU is the I/O CPU. (The I/O CPU specifies which CPU path MME uses to access mainframe memory.)
I/O Disabled	No CPUs are set as the I/O CPU.

Modes

The modes area contains the settings for the Test Mode, Error Mode, and Debug Mode:

- Test Mode can be set to Automatic, Namual, or Compose to indicate testing in automatic, manual, or compose mode.
- Error Mode can be set to <u>stop on channel true</u> to stop testing when a channel error occurs and <u>stop on Sequence true</u> to stop testing when a sequence error (data miscompare) occurs.
- The Debug Mode settings are used in compose mode only.

Compose Mode

Figure 2 shows the MME environment 0 graphical user interface for compose mode. The paragraphs following the figure describe the components of this interface.

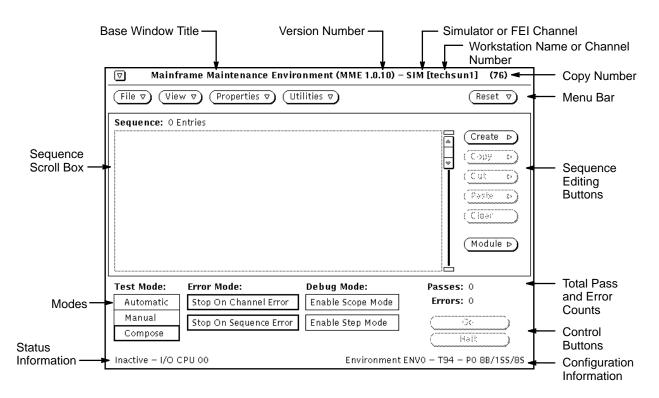


Figure 2. Environment 0 Interface Components (Compose Mode)

Base Window Title

The base window title displays the name of the program: Mainframe Maintenance Environment.

Version Number

The version number indicates the version of MME you are using.

Simulator or FEI Channel

The simulator or FEI channel indicator shows MME is running with the simulator (indicated by SIM) or an FEI channel (indicated by FEI CHN 0 for channel 0, FEI CHN 1 for channel 1, FEI CHN 2 for channel 2, etc.).

Workstation Name or Channel Number

The workstation name or channel number indicator displays the name of the workstation or the channel number where the MME program is running.

Copy Number

The copy number indicates the copy of MME you are using. To set the copy number, start MME with the -copy option. If you start MME with the default copy number of 0, the MME base does not display a copy number. For more information about starting MME with the -copy option, refer to the *MME User Guide*, publication number HDM-102-A.

Menu Bar

The menu bar contains five menu buttons: $(File \nabla)$, $(View \nabla)$, $(Properties \nabla)$, $(Utilities \nabla)$, and $(Reset \nabla)$. For descriptions of the commands accessible from these menu buttons, refer to "Menu Button Commands" on page 10.

Sequence Editing Buttons

Use the sequence editing buttons to manipulate the placement of functions and utilities within a test sequence. These buttons perform the following functions:

Button	Function
Create >	Creates a new function or utility in the current sequence. The function or utility is placed in the Sequence scroll box.
	Choose Create -> Before to place the function before the selected function in the scroll box. Choose Create -> After to place the function after the selected function in the scroll box. Choose Create -> Top to place the function at the top of the scroll box. Choose Create -> Bottom to place the function at the bottom of the scroll box.
Сору Þ	Copies the selected function or utility in the Sequence scroll box. Choose Copy -> Selected to copy the selected function. The Copy -> Range command is not implemented yet.

Button	Function
Cut Þ	Cuts the selected function or utility from the current sequence in the Sequence scroll box. Choose Cut -> Selected to cut the selected function. The Cut -> Range command is not implemented yet.
Paste D	Pastes a copied or cut function or utility into the current sequence in the Sequence scroll box.
	Choose Paste -> Before to paste the function before the selected function in the scroll box. Choose Paste -> After to paste the function after the selected function in the scroll box. Choose Paste -> Top to paste the function at the top of the scroll box. Choose Paste -> Bottom to paste the function at the bottom of the scroll box.
Clear	Removes all functions and utilities from the Sequence scroll box.
(Module D)	Sets the module used by the functions and utilities in the current sequence.

Total Pass and Error Counts

The total pass count component (Passes) indicates the number of passes a test has completed. The total error count component (Errors) indicates the total number of errors found during the current test(s) executions. The pass and error counts are in decimal.

Control Buttons

Use the control buttons to start testing (\bigcirc) and stop testing (\bigcirc Halt).

Configuration Information

The configuration information component displays the current MME environment and configuration data (mainframe type; partition where MME is running; and number of sections, subsections, and banks). The System Configuration Environment (SCE) provides this information.

For example, the following configuration information indicates MME is in environment 0 (Environment ENV0); the mainframe type is a CRAY T94 system (T94); MME is running in partition 0 (P0); and the configuration consists of 8 banks (8B), 1 subsection (1SS), and 8 sections (8S):

Environment ENVO – T94 – PO 8B/1SS/8S

The configuration information component displays one of the following mainframe types:

TesterA tester with 1 CPU and 1 memory modulT94A CRAY T94 mainframeT916A CRAY T916 mainframeT932A CRAY T932 mainframe	ıle

Status Information

The status information component displays the current state of the MME program by using the following messages:

Message	Description
Active	Test(s) are running.
Inactive	No test(s) are running.
I/O CPU ##	The specified CPU is the I/O CPU. (The I/O CPU specifies which CPU path MME uses to access mainframe memory.)
I/O Disabled	No CPUs are set as the I/O CPU.

Modes

The modes area contains the settings for the Test Mode, Error Mode, and Debug Mode:

- Test Mode can be set to <u>Automatic</u>, <u>Nanual</u>, or <u>Compose</u> to indicate testing in automatic, manual, or compose mode.
- Error Mode can be set to <u>Stop in Channel First</u> to stop testing when a channel error occurs and <u>Stop in Sequence First</u> to stop testing when a sequence error occurs.

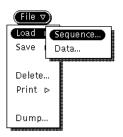
Sequence Scroll Box

The Sequence scroll box shows the function entries for the current sequence. These functions run in the order in which they are displayed in the scroll box; when you click on \bigcirc , function entries at the top of the scroll box run before entries at the bottom.

Menu Button Commands

The menu buttons contain commands that manipulate MME environment 0. This subsection describes what each command does and how to use each menu button command. Figure 9 on page 153 shows all available menu button commands for environment 0.

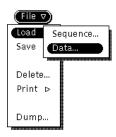
File -> Load -> Sequence



The File -> Load -> Sequence command, as shown at the left, loads previously saved sequences of maintenance channel functions and utilities into the MME program (refer to "File -> Save -> Sequence" for more information about saving sequences). Use this command to load specialized sequences you have created or modified for your specific testing needs. This command displays the MME Load/Save Sequence window:

- 1. Change the directory in the Load Dir field, if necessary, and press the Return key.
- 2. Click on the sequence to load in the Load Files scroll box.
- 3. Click on <u>Load...</u>; MME loads the specified sequence.

File -> Load -> Data



The File -> Load -> Data command, as shown at the left, loads a data set into the MME buffer or mainframe memory. Use this command to re-create a specific data set for testing without having to manually enter it each time. This command displays the MME Load/Save Data window:

Ø MMI	E Load/Save Data
Mode:	Word Parcel
Load Destination:	Memory Buffer
Load Directory:	⊽ usr/*
Load Files:	cfg/ cmd/ diag/ file.sample
Load Type:	Nonstandard
Load Address:	00000000000000000
Load Length:	00000000000000000000 (Load)
Save Source:	Memory Buffer
Save Directory:	usr/
Save File:	file.sample
	000000000000000000000000000000000000000
Save Length:	00000000000000000000000000000000000000
	9 files found

Perform the following procedure to manipulate this window:

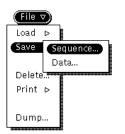
- 1. Select the data Mode. Click on wird to use a word load address and length or read to use a parcel load address and length. The Load Address and Load Length fields change to the specified format.
- 2. Specify the Load Destination. Click on weiney to load the data into mainframe memory or Bulfer to load the data into the MME buffer.
- 3. Change the directory, if necessary, by choosing a new directory from the Load Directory: ⊽ or by entering the directory in the Load Directory field.

The Load Directory: includes selections for the usr diagnostic and data directories, all of the rel directories, and all of the alpha directories.

4. In the Load Files scroll box, click on the data file you want to load. The Load Type and Load Length information is updated, and the Load button is activated.

- 5. In the Load Address field, enter the address at which you want to load data.
- 6. Click on Load); MME loads the data at the specified address in mainframe memory or the MME buffer.

File -> Save -> Sequence

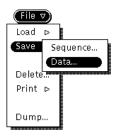


The File -> Save -> Sequence command, as shown at the left, saves a sequence of maintenance channel functions and utilities you have created or modified. Use this command to create customized sequences (refer to "File -> Load -> Sequence" earlier in this section for more information about loading sequences). This command displays the MME Load/Save Sequence window:

🖉 MME Load/Save Sequence
Load Dir: <u>usr/seq/*</u> Load Files:
config.Z pcip.Z sanity.Z sanity.test.Z tester.config.Z tester.config.dma.Z
(L(ad)
Save Dir: <u>usr/seq</u> Save File: <u></u> Save

- 1. To change the directory on the maintenance workstation (MWS) where the sequence is saved, specify a different directory in the Save Dir field, and press the Return key.
- 2. In the Save File field, enter the name of the file you want to use, and press the Return key.
- 3. Click on <u>save...</u> or press the Return key; MME saves the sequence in the specified file.

File -> Save -> Data



The File -> Save -> Data command, as shown at the left, saves a mainframe memory data set or an MME buffer data set so you can reuse it. Use this command to re-create a specific data set for testing. This command displays the MME Load/Save Data window:

Ø MM	E Load/Save Data
Mode:	Word Parcel
Load Destination:	Memory Buffer
Load Directory:	⊽ usr/*
Load Files:	cfg/
	· · · · · · · · · · · · · · · · · · ·
	layout/
Load Type:	Plain File
	000000000000000000000000000000000000000
Load Length:	(concorrection)
Save Source:	Memory Buffer
Save Directory:	usr/
Save File:	
Save Address:	000000000000000000000000000000000000000
Save Length:	00000000000000000000000000000000000000
Checking directory	8 files found

- 1. Click on Mode: wide to use a word save address and length or Mode: Mo
- 2. Click on Save Source: Neuropy to save mainframe memory data, or click on Save Source: Eulfer to save MME buffer data.
- 3. To change the directory where the data is saved on the MWS, specify a different directory in the Save Dir field, and press the Return key.
- 4. In the Save File field, enter the name of the file you want to use.
- 5. In the Save Address field, enter the starting address of the data block you want to save.
- 6. In the Save Length field, enter the length of the data block you want to save.
- 7. Click on (Save); MME saves the specified data set.

File -> Delete



The File -> Delete command, as shown at the left, deletes files you no longer need. Use this command to delete unwanted files from the MME user directories stored on the MWS. This command displays the MME Delete File window:

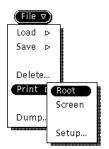
🖉 🛛 MME Delete File	
Dir: ▼ usr/seq/*	_
Files:	
Vread.ckva.Z a1.Z	
brdwt.Z	P.
break.Z	
cfgtmp.Z	
cft.seq.Z	
cft.seq1.Z	
cicjtest.Z	
cicjtest2.Z	
config.1x4.Z	
configtest.Z	
contest.Z	
dbchk.Z	
diag.Z	
81 files	found

- 1. Change the directory, if necessary, by:
 - Entering the directory in the Dir.) field and pressing the Return key, or
 - Choosing the directory from the Dir. v) button. The following user directories are available:

Directory	Description
usr/data	User data files
usr/diag	User diagnostic programs
usr/env	User environment files
usr/layout	User layout data
usr/lst	User listings
usr/seq	User sequences

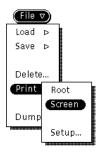
- 2. Click on the file you want to delete.
- 3. Click on **Later**; MME deletes the file.

File -> Print -> Root



The File -> Print -> Root command, as shown at the left, prints an image of everything contained in the root window, including the MME base window.

File -> Print -> Screen

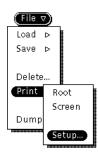


The File -> Print -> Screen command, as shown at the left, prints an image of a window or an icon.

When you choose this command, the cursor becomes a plus symbol (+). Move the cursor to the window or icon you want to print, and click any mouse button.

NOTE: This command does not print an image of the MME base window. To print an image of the MME base window, use the File -> Print -> Root command.

File -> Print -> Setup



The File -> Print -> Setup command, as shown at the left, enables you to edit the commands that control how MME prints data for the File -> Print -> Root and File -> Print -> Screen commands. This command displays the MME Print Setup window:

Q	MME Print Setup
	Print Root Command: I root xpr -scale 3 -device ps -gray 3 /bin/lp)& Print Screen Command: I me xpr -scale 3 -device ps -gray 3 /bin/lp)& Print Text Command: Ip
	Save Commands To File Reset Commands From File Reset Commands From Defaults

Modify the commands in the Print Root Command and Print Screen Command fields to change how MME prints. For more information about the UNIX xwd, xpr, and lp commands used in the print processes, refer to the UNIX online manual (man) pages (enter **man xwd**, **man xpr**, or **man lp** at a UNIX command prompt). Use the buttons in the MME Print Setup window to perform the following functions:

- To save the current printer setup commands for later use, click on Save Commands To File.
- To load the printer setup commands you saved previously, click on (Reset Commands From File).
- To load the default printer setup commands that the MME program provides, click on (Reset Commands From Defaults).

File -> Dump



The File -> Dump command, as shown at the left, sends a copy of MME buffer or mainframe memory data to a file or printer. Use this command to create a permanent record of the data so you can analyze it later. This command displays the MME Dump Setup window:

Q	MME D	ump Setup							
Mode:									
File	Printer	Compress							
Directe	nyo <u>usiv</u>								
Ŵ	lle:								
Form	i at: 🔽 Pa	arcel							
Sour	ce: Mem	ory Buffer							
04t									
		000000000000000000000000000000000000000							
-		000000000000000000000000000000000000000							
Commer	its:								
*									
		*							
	(Dump)								

Perform the following procedure to manipulate the MME Dump Setup window:

1. Specify the Mode. Click on **File** to output the data to a file, or click on **Prime** to output the data to the printer.

- **NOTE:** Click on <u>Compress</u> to compress the output of the File -> Dump command. This reduces the size of the data listing by replacing repeated lines with a statement similar to Last line repeated 077 (63) times.
- 2. If data is being dumped to a file, specify the directory in the Directory field and the file in the File field.
- 3. Choose the format in which you want the data displayed from the Format: . The following formats are available:

Format	Description
Nibble	Nibble data
Byte	Byte data
Parcel	Parcel data
Halfword	Halfword data
Word	Word data
Hexadecimal	Hexadecimal data
Address	Address data
Text	Text data
Exchange (NI)	Exchange data (not implemented)
Instruction (T90)	Instruction data (CRAY T90 series instructions)
Instruction (IEEE)	Instruction data (IEEE instructions)
Instruction (C90)	Instruction data (CRAY C90 series instructions)

- 4. Specify the Source. Click on weinty to dump mainframe memory data or Buller to dump MME buffer data.
- 5. Specify the data block you want to dump by completing any two of the following actions (the third field is automatically set using data from the other two fields):
 - Enter the starting address of the data block you want to dump in the Start Address field, and press the Return key.
 - Enter the length (in words) of the data block in the Length (words) field, and press the Return key.
 - Enter the ending address of the data block you want to dump in the End Address field, and press the Return key.
- 6. Click on the Comments scroll box and type any comments you want to include with the data. For example, you might specify that the data was created on a specific date.
- 7. Click on <u>Dump</u>; MME sends the data to the printer or file.

View -> Memory



The View -> Memory command, as shown at the left, displays mainframe memory in a separate window. Use this command to verify the contents of specific mainframe memory locations or to change the data stored in mainframe memory. This command displays the MME View Memory Setup window:

Ø MM	9 MME View Memory Setup										
Refresh (msec): 1000 🛛 🗕 🗁 🛛											
Mode:	Mode:										
Memory	Exchar	ige	Instruction								
Format:											
Nibble	Halfw	ord/	Text								
Byte	Word		Address								
Parcel	Hex										
Source:											
Memo	ory		Buffer								
Size:	Mediur	n w	 D								

Perform the following procedure to manipulate this window:

- 1. Set the interval at which memory windows are updated by moving the Refresh (msec) slider or by entering a value in the Refresh (msec) field and pressing the Return key. (Setting this value too low can monopolize the workstation CPU and reduce system performance.)
- 2. Click on a Mode (<u>Nemey</u>, <u>Ecchange</u>, or <u>Instruction</u>) to specify the way you want the data displayed.
- Click on a Format. The following Format settings are available for mode: <u>Nibble</u>, <u>Halfword</u>, <u>Text</u>, <u>Byte</u>, <u>wird</u>, <u>Address</u>, <u>Parcel</u>, and <u>Hes</u> (hexadecimal).

The following Format settings are available for Exchange mode: [Byte], [wird], [Partel], [Text], [Halfword], and [Address].

The following Format settings are available for <u>Instruction</u> mode: [50, T30], and IEEE. Memory mode (<u>Neuron</u>) displays normal memory:

Q	Memo	ry – Ab	solute	
000000000000	000000	000000	000000	000000
0000000000000	000000	000000	000000	000000
000000000002	000000	000000	000000	000000
000000000003	000000	000000	000000	000000
000000000000	000000	000000	000000	000000
000000000005	000000	000000	000000	000000
000000000006	000000	000000	000000	000000
000000000007	000000	000000	000000	000000
000000000010	000000	000000	000000	000000
000000000011	000000	000000	000000	000000
000000000012	000000	000000	000000	000000
000000000013	000000	000000	000000	000000
000000000014	000000	000000	000000	000000
00000000015	000000	000000	000000	000000
000000000016	000000	000000	000000	000000
000000000017	000000	000000	000000	000000
000000000020	000000	000000	000000	000000
000000000021	000000	000000	000000	000000
000000000022	000000	000000	000000	000000
00000000023	000000	000000	000000	000000
00000000024	000000	000000	000000	000000
00000000025	000000	000000	000000	000000
00000000026	000000	000000	000000	000000
00000000027	000000	000000	000000	000000
00000000030	000000	000000	000000	000000
00000000031	000000	000000	000000	000000
00000000032	000000	000000	000000	000000
00000000033	000000	000000	000000	000000
00000000034	000000	000000	000000	000000
00000000035	000000	000000	000000	000000
00000000036	000000	000000	000000	000000
00000000037	000000	000000	000000	000000

Exchange mode (techange) displays exchange information:

Q								Memo	iry	- Abso	lu	te						
CN 0	000000 000000 000014 000014 000014 000014 000014 000014 000014	4100 40 40 40 40 40 40		0 00 1 00 2 00 3 00 4 00 5 00 5 00 7 00	0000 0000 0000 0000 0000 0000 0000 0000 BR	0 00 0 00 0 00 0 00 0 00 0 00 0 00 8 11		00000 00000 00000 00000 00000 00000 0000	10 10 10 10 10	000000 000100 000100 000000 000000 000000	51 52 53 54 55 56 57 II		77 2 00 (00 (00 (00 (00 (00 (FII	00000 17777 00000 00000 00000 00000 00000 00000 0000	7 1 10 0 10 0 10 0 10 0 10 0 10 0 11 III	00000 77777 00000 00000 00000 00000 00000 00000 0000	173 00(00(00(00(00(00(111	2000 7777 2000 2000 2000 2000 2000 2000
VL 2	200 FUS 00		ODN VVV SSS	000	D1	0E 1 **V **N	FWB PSM S L	IF	00	000000	RM PE	PRR FOP	EBM EPE		IPD OCL IPD OCL II	IXM MNA	XNU IXX XNU IXN F	ODN VVV ODN VIV FVI
LATO LATI LAT2 LAT3 LAT4 LAT5 LAT6 LAT7	I RWXC 2 RWXC 3 RWXC 4 RWXC 5 RWXC 6 RWXC	00 00 00 00 00 00	RWMD RWXD RWXD RWXD RWXD RWXD RWXD RWXD	00 00 00 00 00 00	PB PB PB PB PB PB	0000 0000 0000 0000 0000 0000	0000(0000(0000(0000(0000(0000(00000 00000 00000 00000 00000 00000 0000	LB LB LB LB LB LB LB)00()00()00()00()00(00000 00000 00000 00000 00000 00000	0 LI 0 LI 0 LI 0 LI 0 LI	L 000 L 000 L 000 L 000 L 000 L 000		77740 00000 00000 00000 00000 00000 00000 0000	000 000 000 000 000 000	

MME displays the same data for CRI floating-point format exchange packages and IEEE floating-point format exchange packages. The Memory window contains information for both formats. MME highlights only the bits and modes that are used for the CP modules included in the mainframe. This document shows exchange packages for CP modules that use the CRI floating-point format.

For more information about the contents of CRI floating-point format exchange packages, refer to the *CPU Module (CP02)* document, publication number HTM-003-0. For more information about the contents of IEEE floating-point format exchange packages, refer to the *IEEE CPU Module (CP02)* document.

Instruction mode (

Q		Memory – Absolute
00000000000a	000000	ERR
0000000000b	000000	ERR
0000000000c	000000	ERR
00000000000d	000000	ERR
0000000001a	000000	ERR
00000000001b	000000	ERR
0000000001c	000000	ERR
0000000001d	000000	ERR
0000000002a	000000	ERR
0000000002b	000000	ERR
0000000002c	000000	ERR
0000000002d	000000	ERR
0000000003a	000000	ERR
0000000003b	000000	ERR
0000000003c	000000	ERR
0000000003d	000000	ERR
0000000004a	000000	ERR
0000000004b	000000	ERR
0000000004c	000000	ERR
0000000004d	000000	ERR
0000000005a	000000	ERR
0000000005b	000000	ERR
0000000005c	000000	ERR
0000000005d	000000	ERR
0000000006a	000000	ERR
0000000006b	000000	ERR
0000000006c	000000	ERR
0000000006d	000000	ERR
0000000007a	000000	ERR
0000000007b	000000	ERR
0000000007c	000000	ERR
0000000007d	000000	ERR

4. Choose the size of the display window from the Size: . This affects only the memory and instruction mode windows. The following window sizes are available:

Size	Description
Small	The window displays 108 words.
Medium	The window displays 20_8 words.
Large	The window displays 40_8 words.
X–Large	The window displays 100 ₈ words.

5. Choose the font size you want to display in the window from the Font: 🖸. The following font sizes are available:

Size	Description
Small	The font size is small.
Medium	The font size is medium.
Large	The font size is large.
X–Large	The font size is extra large.

- 6. Change the starting address, if necessary, by double clicking on the Address field and typing a new value.
- 7. Click on <u>view</u>. MME displays a Memory Absolute window for the specified memory location.

If you want to change the Format, Memory, Exchange, Instruction, Window Size, or Window Font from the Memory – Absolute window, press the MENU mouse button and choose the menu item:

ſ	Q		Memo	ry – Ab	solute	
I	0000000		00000	000000	000000	000000
I	0000000	0001	000000	000000	000مم	000000
I	0000000	Form	nat		⊳ ∎000	000000
I	0000000				₽ 00	000000
I	0000000				D00	000000
I	0000000	Mem	tory (Met	ta-M)	₽ 00	000000
I	0000000	/ Euch	ongo (M	ata_V)	D 00	000000
I	0000000	LEXCH	ange (M	eta=x)	D [000	000000
I	0000000	Instr	uction (I	Meta-I)		000000
I	0000000				D 00	000000
I	0000000				D00	000000
I	0000000	Wind	low Size		⊳ ∎00	000000
I	0000000	1.12			D00	000000
I	0000000		dow Font	-	⊳ þ oo	000000
I						000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
I	0000000		000000	000000	000000	000000
	0000000	0037	000000	000000	000000	000000

HDM-008-A

For example, the following Memory – Absolute window appears if you choose the Exchange (Meta–X) format menu option:

Q								Memo	iry	– Abso	plut	te						
ADX	00000	1000	0000	0														
P	000004				0000	0 00	0000	00000	0	000000	S0	0000	000 (00000	0 0	00000	000	0000
PN	000		A	1 01	0000	0 00	0000	00000	0	000100	S1	1777	'77 ·	17777	'7 1i	77777	177	777
XA	000014				0000	~ ~~	0000	00000		000100	S2	0000		00000		00000		0000
EA0	000014				0000		0000	00000		000000	53	0000		00000		00000		0000
EA1	000014	· •	A.		0000		0000	00000		000000	54	0000		00000		00000		0000
EA2			A		0000	~ ~~	0000	00000		000000	S5 S6	0000		00000 00000	~ ~	00000		0000
EA3			Ä		0000		0000	00000		0000000	55	0000		10000		10000		0000
I c.m	00001-	10		0	0000	0 00	0000	00000	0	000000	37	0000	,000	00000	0 0	00000	000	0000
ICN 1	000	MO	DDES	015	BR	RS U	EBM	ΤM	00	000000	ΤT	TTT	FTT	TTT	TTT	IFT	TTT	TTT
	200						SDIE				RŨ	FOP	EBC	MRI	IPD	MNA	XNU	ODN
					D1	0E	LM				ΡM	PRR	XPM	CTP	OCL	IXM	IXX	VVV
STA	TUS OO	XNU					FWB	IF	00	000000	RM	FOP	EBM		IPD		XNU	ODN
			1 VVV		BB		PSM				PE		EPE	~ . ~	OCL.			VIV
		555	5 555		MU	**U	Sι				ΕU	EEE	XIC	UTN	II	IXI	F	FVI
LATI	O RWKOC	02	R₩ Ø D	02	PB	nnnn	nnnni	00000	LB	000000	າດດເ	າດດດດ	n i	177	777	77740	INNN	
LLAT			RWXD					10000	LB							honon		
LAT:	2 RWXC	ŌŌ	RWXD	ŌŌ	ΡB	0000	0000	00000	ĹΒ		0000	00000	DO L	L 000	1000	00000	000	
LAT	3 RWXC	00	RWXD	00	PB	0000	0000	00000	LB							00000		
LAT4		00	RWXD					00000	LB							00000		
LAT		00	RWXD	~~		~~~~	~~~~	00000	LB							00000		
LAT		00	RWXD					00000	LB							00000		
LAT	7 RWXC	00	RWXD	00	PB	0000	0000	00000	LB	000000	1000	10000	JU L	L UOC	10001	00000	000	

You can also change the window format (Format), data type (Memory or Instruction), window size (Window Size), or window font (Window Font) from this menu.

In this example, instead of using the MENU mouse button, you may also use the diamond-shaped meta key (or , depending on the type of keyboard you have) with one of several keyboard shortcuts. The following key combinations are shortcuts to the menu options:

Key Sequence	Function
Meta-a	Switches the display to address format
Meta-n	Switches the display to nibble format
Meta-b	Switches the display to byte format
Meta-p	Switches the display to parcel format
Meta-h	Switches the display to halfword format
Meta-w	Switches the display to word format
Meta-e	Switches the display to hexadecimal format
Meta-t	Switches the display to text format
Meta-i	Switches the display to instruction mode
Meta-x	Switches the display to exchange mode
Meta-m	Switches the display to memory mode

Changing Memory

Perform the following procedure from the Memory – Absolute window to change data stored in mainframe memory:

1. Use the arrow keys to move the cursor to the location in memory you want to change, or click on the location. In this example, parcel 0000000005b was selected:

ſ	Q	Memo	ry – Ab	solute	
	00000000000000	000000	000000	000000	000000
	000000000001	000000	000000	000000	000000
	000000000002	000000	000000	000000	000000
	000000000003	000000	000000	000000	000000
	000000000004	000000	<u>0</u> 00000	000000	000000
	000000000005	000000	00000	000000	000000
	000000000006	000000	000000	000000	000000
	000000000007	000000	000000	000000	000000
	000000000010	000000	000000	000000	000000
	000000000011	000000	000000	000000	000000
	00000000012	000000	000000	000000	000000
	00000000013	000000	000000	000000	000000
	00000000014	000000	000000	000000	000000
	00000000015	000000	000000	000000	000000
	00000000016	000000	000000	000000	000000
	00000000017	000000	000000	000000	000000
	00000000020	000000	000000	000000	000000
	00000000021	000000	000000	000000	000000
	00000000022	000000	000000	000000	000000
	00000000023	000000	000000	000000	000000
	00000000024	000000	000000	000000	000000
	00000000025	000000	000000	000000	000000
	00000000026	000000	000000	000000	000000
	00000000027	000000	000000	000000	000000
	00000000030	000000	000000	000000	000000
	00000000031	000000	000000	000000	000000
	00000000032	000000	000000	000000	000000
	00000000033	000000	000000	000000	000000
	00000000034	000000	000000	000000	000000
	00000000035	000000	000000	000000	000000
	00000000036		000000	000000	000000
l	00000000037	000000	000000	000000	000000

2. Type the new value you want to place in the memory location. The entire word is highlighted, which enables you to change it. Use the arrow keys to move through the highlighted word.

In the following example, 000217 was typed at memory location 0000000005b:

Q	Ø Memory – Absolute							
000000000000	000000	000000	000000	000000				
000000000001	000000	000000	000000	000000				
00000000002	000000	000000	000000	000000				
00000000003	000000	000000	000000	000000				
000000000004	000000	000000	000000	000000				
000000000005	000000	000217	000000	000000				
00000000000	000000	000000	000000	000000				
00000000007	000000	000000	000000	000000				
00000000010	000000	000000	000000	000000				
00000000011	000000	000000	000000	000000				
00000000012	000000	000000	000000	000000				
00000000013	000000	000000	000000	000000				
00000000014	000000	000000	000000	000000				
00000000015	000000	000000	000000	000000				
00000000016	000000	000000	000000	000000				
00000000017	000000		000000	000000				
00000000020	000000	000000	000000	000000				
00000000021	000000	000000	000000	0000000				
00000000022	000000	000000	000000	0000000				
00000000023	000000	000000	000000	0000000				
00000000025	000000	000000	000000	0000000				
00000000025	000000	000000	000000	0000000				
000000000027	000000	000000	000000	0000000				
00000000030	000000	000000	000000	000000				
00000000031	000000	000000	000000	000000				
00000000032	000000	000000	000000	000000				
00000000033	000000	000000	000000	000000				
00000000034	000000	000000	000000	000000				
00000000035	000000	000000	000000	000000				
00000000036	000000	000000	000000	000000				
00000000037	000000	000000	000000	000000				

3. Press and release the Return key to update memory. This example shows memory location 0000000005b changed from 000000 to 000217:

	Q	· ·							
	000000000000	000000	000000	000000	000000				
	00000000000000001	000000	000000	000000	000000				
I	0000000000000002	000000	000000	000000	000000				
	000000000003	000000	000000	000000	000000				
I	000000000000	000000	000000	000000	000000				
I	0000000000000005	000000	000217	000000	000000				
I	000000000006	000000	000000	000000	000000				
I	000000000000	000000	000000	000000	000000				
I	000000000010	000000	000000	000000	000000				
I	00000000011	000000	000000	000000	000000				
I	00000000012	000000	000000	000000	000000				
I	00000000013	000000	000000	000000	000000				
I	00000000014	000000	000000	000000	000000				
	00000000015	000000	000000	000000	000000				
	00000000016	000000	000000	000000	000000				
	00000000017	000000	000000	000000	000000				
	000000000020	000000	000000	000000	000000				
	000000000021	000000	000000	000000	000000				
	000000000022	000000	000000	000000	0000000				
	000000000023	000000	000000	000000	0000000				
	000000000025	000000	000000	000000	0000000				
	000000000026	000000	000000	000000	0000000				
I	000000000027	000000	000000	000000	0000000				
	000000000030	000000	000000	000000	000000				
	00000000031	000000	000000	000000	000000				
	00000000032	000000	000000	000000	000000				
	00000000033	000000	000000	000000	000000				
	00000000034	000000	000000	000000	000000				
	00000000035	000000	000000	000000	000000				
	00000000036	000000	000000	000000	000000				
	00000000037	000000	000000	000000	000000				

4. Repeat Steps 1 through 3 to change additional memory locations.

Using the Keyboard Accelerator

The keyboard accelerator offers another way to change memory and includes several other features. To access the keyboard accelerator, move the cursor inside a memory (or buffer) window and press the spacebar. The MME Keyboard Accelerator window appears:



Dump Command

The first command in the MME Keyboard Accelerator window is the Dump command. The Dump command dumps data to a file or the printer. When you type the letter D in the MME Keyboard Accelerator window, the window changes to:

Q	MME Keyboard Accelerator
Address:	[0-7]* - SPACE when complete
Dump Buf	fer to Printer Parcel

NOTE: By default, the Dump command dumps buffer data to the printer. To change this, press the Back Space key several times to delete the default selections. Then, type the menu options you want.

Enter the starting address of the data block you want to dump and press the spacebar; for example, if you wanted to dump the data block starting at 100, you would enter **100**:

Q	MME Keyboard Accelerator
Length: [0-7]	* - RETURN when ready
Dump Buffer t	co Printer Parcel 100

Then, enter the length of the data block you want to dump; for example, if you wanted to dump a block of 2000_8 parcels, you would enter **2000**:

Q	MME Keyboard Accelerator						
Length:	[0-7]* - RETURN when ready						
Dump Bu	ffer to Printer Parcel 100 2000						

Finally, press the Return key to dump the data. The window displays the main menu again:

Q	MME Keyboard Accelerator								
Commands:	Dump	Enter	G O	Halt	Load	S ave	0-7	RETURN	

Enter Command

The second command in the MME Keyboard Accelerator window is the Enter command. The Enter command puts data into memory. When you type the letter E in the MME Keyboard Accelerator window, the window changes to:

Q	MME Keyboard Accelerator						
Enter: Enter	e X change,	Auto,	address	[0-7]*[a-d]			

This window gives you three command options:

• Type the letter **X** in the Enter menu to enter exchange data; *this option is not implemented yet*:

Q	MME Keyboard Accelerator
Not implemented Enter eXchange	- RETURN when ready

• Type the letter **A** in the Enter menu to start the automatic increment mode. Automatic increment mode enables you to enter data into consecutive memory locations without having to manually enter each memory address.

For example, if you wanted to enter data in consecutive memory locations starting at address 102, you would perform the following steps:

1. Enter the address at which you want to enter the data and press the Return key. For this example, you would enter **102**:

Q	MME Keyboard Accelerator							
Data:	RETURN when complete, ESC to cancel							
Enter	Auto 102							

2. Enter the first item of data and press the Return key. For this example, you would enter **17777 177777**:

Q	MME Keyboard Accelerator
Data:	RETURN when complete, ESC to cancel
Enter	Auto 102 17777 17777

The window advances to the next memory location:

Q	MME Keyboard Accelerator
Data:	RETURN when complete, ESC to cancel
Enter	Auto 103

3. Enter the next item of data and press the Return key. Repeat this process to enter all of your data. When you have finished entering data, press the Esc key to halt automatic increment mode. The MME Keyboard Accelerator window returns to the main menu:

 Ø
 MME Keyboard Accelerator

 Commands: Dump Enter Go Halt Load Save 0-7 RETURN

• Type a number-letter combination in the Enter menu to enter a parcel address that indicates the memory address you want to change. For example, if you wanted to change parcel 1000b, you would enter **1000b**:

Q	MME Keyboard Accelerator
Address: Enter 10	[O-7]*[a-d] - SPACE when complete

Then, enter the data you want to write to memory; for example, if you wanted to enter 177777 177777 177777 177777, you would enter **177777 177777 177777** 177777

Q		MME H	Keyboar	rd Accelerator	
RETURN when Enter 1000b	ready 177777	177777	177777	177777	
Encor 1000b					

Press the Return key to write the data to memory:

ſ	Q	Memory – Absolute								
l	00000001000	000000	177777	177777	177777					
L	00000001001	177777	000000	000000	000000					
L	00000001002	000000	000000	000000	000000					

The window displays the main menu again:

Q	MME Keyboard Accelerator Dump Enter Go Halt Load Save 0-7 RETURN								
Commands:	Dump	Enter	Go	Halt	Load	Save	0-7	RETURN	

Go Command

The third command in the MME Keyboard Accelerator window is the Go command. The Go command runs the selected test sequences. When you type the letter **G** in the MME Keyboard Accelerator window, the window changes to:

Q	MME Keyboard Accelerator
RETURN when ready Go ∎	

Press the Return key to start the selected test sequences. The window displays the main menu again:

ľ	Q	MME Keyboard Accelerator								
	Commands:	Dump	Enter	Go	Halt	Load	Save	0-7	RETURN	

Halt Command

The fourth command in the MME Keyboard Accelerator window is the Halt command. The Halt command halts the currently running test sequences. When you type the letter H in the MME Keyboard Accelerator window, the window changes to:

Q	MME Keyboard Accelerator
RETURN when ready Halt	

Press the Return key to halt the test sequences. The window displays the main menu again:

	Q				E Key					
C	ommands:	Dump	Enter	Go	Halt	Load	S ave	0-7	RETURN	

Load Command

The fifth command in the MME Keyboard Accelerator window is the Load command. The Load command loads a data set or sequence. When you type the letter L in the MME Keyboard Accelerator window, the window changes to:

Q	MME Keyboard Accelerator
Load: D ata, Load	, S equence

This window gives you two options:

- Type the letter **D** in the Load menu to load a data set; *this command is not implemented yet.*
- Type the letter **S** in the Load menu to load a sequence; *this command is not implemented yet*.

Save Command

The sixth command in the MME Keyboard Accelerator window is the Save command. The Save command saves a data set or sequence. When you type the letter **S** in the MME Keyboard Accelerator window, the window changes to:

Q	MME Keyboard Accelerator
Save: Data, Save	, Sequence

This window gives you two options:

- Type the letter **D** in the Save menu to save the current data set; *this command is not implemented yet.*
- Type the letter **S** in the Save menu to save the current sequence; *this command is not implemented yet.*

Numeric Commands

Two options are available when you type a number in the MME Keyboard Accelerator window: you can display memory starting at the number, or you can convert the number from octal to parcel format or from parcel format to octal.

• To display a specific memory location, type the location in the MME Keyboard Accelerator window and press the Return key. For example, to view memory location 5000, enter **5000**:

Q		/board Acc		
[0-7]*[a-d], RETURN 5000	to chang	e display,	SPACE to convert	

The Memory — Absolute window displays memory at location 5000:

0	9 Memory – Absolute						
00000005000	000000	000000	000000	000000			
00000005001	000000	000000	000000	000000			
00000005002	000000	000000	000000	000000			
00000005003	000000	000000	000000	000000			
00000005004	000000	000000	000000	000000			
00000005005	000000	000000	000000	000000			
00000005006	000000	000000	000000	000000			
00000005007	000000	000000	000000	000000			
00000005010	000000	000000	000000	000000			

The window displays the main menu again:

 Ø
 MME Keyboard Accelerator

 Commands: Dump Enter Go Halt Load Save 0-7 RETURN

• To convert a number from octal to parcel format or from parcel format to octal, type the number and press the spacebar. For example, to convert octal value 5000 to parcel format, enter **5000** and press the spacebar:



The parcel format equivalent is displayed:

Q	MME Keyboard Accelerator
RETURN when done 5000 = 01200a	

Press the Return key, and the window displays the main menu again:

Q		Ν	4MI	E Key	board	l Acce	elera	tor	
Commands:	Dump	Enter	Go	Halt	Load	Save	0-7	RETURN	

Return

To close the MME Keyboard Accelerator window, press the Return key while the cursor is in the window.

View -> Buffer



The View -> Buffer command, as shown at the left, displays MME buffer data in a separate window. Use this command to verify the contents of specific MME buffer memory locations or to change the data stored in the MME buffer. This command displays the MME View Memory Setup window:

MME View Memory Setup								
Refresh (msec): <u>1000</u> [] — [] — []								
	Mode:							
Memory	Exchar	ge	Instruction					
Format:								
Nibble	Halfw	ord	Text					
Byte	Word		Address					
Parcel	Hex							
Source:								
Memo	ory		Buffer					
Size: ☐ Large Font: ☐ Medium Address: View								

This is the same window that appears when you choose View -> Memory, except <u>bulker</u> is set as the Source, indicating MME will display MME buffer memory in a window. For more information about manipulating the MME View Memory Setup window, changing data stored in memory, and using the keyboard accelerator; refer to "View -> Memory" on page 18.

NOTE: When you are viewing MME buffer data, the window header displays Buffer instead of Memory – Absolute:

-					
	0		Buffer		
	000000000000	000000	000000	000000	000000
	000000000001		000000		
	000000000002	000000	000000	000000	000000
	00000000003		000000		
	000000000004		000000		
	000000000000		000000		
	000000000006		000000		
I	00000000007	000000	000000	000000	000000

View -> Log



The View -> Log command, as shown at the left, displays the MME Log window:

Running DMA Chip test - Pattern = ODD BITS Mrite CPU = 0, Read CPU = 0	
Running DMA Chip test – Pattern = ODD BITS Yrite CPU = 1, Read CPU = 1	
Running DMA Chip test - Pattern = ODD BITS Mrite CPU = 2, Read CPU = 2	
Running DMA Chip test - Pattern = ODD BITS Mrite CPU = 3, Read CPU = 3	
Running DMA Chip test - Pattern = EVEN BITS Mrite CPU = 0, Read CPU = 0	
Running DMA Chip test - Pattern = EVEN BITS Mrite CPU = 1, Read CPU = 1	
Running DMA Chip test - Pattern = EVEN BITS Mrite CPU = 2, Read CPU = 2	
Running DMA Chip test - Pattern = EVEN BITS Mrite CPU = 3, Read CPU = 3	
Running DMA Chip test - Pattern = WADDR Mrite CPU = 0, Read CPU = 0	
Running DMA Chip test - Pattern = WADDR Write CPU = 1, Read CPU = 1	

Use this command to view any errors that occur while a test is running.

View -> Report



The View -> Report command, as shown at the left, displays the error report information in the MME Report Display window. Use this command to view the error report generated by the memory or instruction buffer tests. This command displays the MME Report Display window:

Ø MME Report Display	
View: Differences Only (Clear Report)	
Offset Expected (+000000) Actual (+000000) Difference (+000000)	
No Report Data Available.	

This window displays the Expected, Actual, and Difference values for the contents of memory where differences are detected. These values are offset into the MME buffer, which is indicated by the value in parentheses as shown in the previous MME Report Display.

Use the Little setting and Clear Report) button in this window to:

- Display only the addresses where differences occurred ([Lilfarances]).
- Clear the MME Report Display window of data ((Clear Report)).

View -> Notes



The View -> Notes command, as shown at the left, displays the MME release notes in a separate window. Use this window to read about any changes to MME for the current offline diagnostic release.

Properties -> Environment -> ENV1



The Properties -> Environment -> ENV1 command, as shown at the left, switches MME to environment 1. Use this command to switch to single-control-point testing. Refer to the "Environments 1 and 2" section later in this document for more information about environment 1.

Properties -> Environment -> ENV2



The Properties -> Environment -> ENV2 command, as shown at the left, switches MME to environment 2. Use this command to switch to multiple-control-point testing. Refer to the "Environments 1 and 2" section later in this document for more information about environment 2.

Properties -> Partition



The Properties -> Partition command, as shown at the left, selects the logical partition in which MME will run. Use this command to select the partition you want to troubleshoot. MME scans the current configuration for available partitions and allows you to select only partitions that allow maintenance or concurrent maintenance. MME displays the available partitions in a menu attached to the Properties -> Partition menu command.

Properties -> Resource Allocation



The Properties -> Resource Allocation command, as shown at the left, changes the way MME performs. Use this command to specify which CPU is used to write and read memory, which spare-chip table that MME uses, and which debug level MME uses. This command displays the MME Resource Allocation window; choose the category you want to modify from the Category: \bigtriangledown . (Only the CPU Allocation/Control, Spare Chips, and Miscellaneous menu options have parameters that apply to environment 0.)

NOTE: The Control Point Properties option also brings up an active parameter, but this parameter is not valid in environment 0. Do not change this parameter (Maximum Pass) in environment 0. Future versions of MME will not include this parameter in environment 0.

Specifying Which CPU Writes and Reads Memory

To specify which CPU writes to and reads from mainframe memory, choose **CPU Allocation/Control** from the Category: ☑. The MME Resource Allocation window changes to:

S MME Res	ource Allo	cation					
Catagory: 🔽 CPU Allocation/Control							
Auto Assignment: Disabled Enabled							
CPU Mode:	 3						
I/O CPU	00		<u></u>				
	01	1 21	31				
	02	2 22	32				
	03	3 23					
	134	4 24	34				
	05 1	5 25	35				
	08.11	8 28	38				
	07 1	7 27	37				
	ها است (`	Teggio)	£8				
PCI:							

Click on the CPU that you want to write to and read from memory.

Modifying the Spare-chip Table that MME Uses

To modify the spare-chip table that MME uses, choose **Spare Chips** from the Category: \bigtriangledown . The MME Resource Allocation window changes to:

Q	MME Resource Allocation		
Catagory: (▼ Spare Chips		
Currently the sparechip table contains the only the configured spares.			
Mode:	Reset – Use Configured Spares		
	Merge With Configured Spares		
	Overide Configured Spares		
Upper Bad Bit Code: () ((((-Bits 48 0 3)) Lower Bad Bit Code: () ((((-Bits 16 0)) Apply Selections)			

Use this window to modify the spare-chip table that MME uses. This enables you to test the mainframe with different configurations of spare memory chips.

To use this window, perform the following procedure:

1. Click on a Mode. The Mode settings perform the following functions when you click on (Apply Selections):

Setting	Description
Reset – Use Confinued Suales	MME writes a spare-chip table to the mainframe that contains only the spare chips configured in SCE.
Norge with Conligued Spares	MME writes a spare-chip table to the mainframe that contains only the bad bit codes that you specify with the Upper Bad Bit Code: ☑ and Lower Bad Bit Code: ☑ menus. (This setting overrides the spare-chip selections in SCE.)

Setting

Overide Configured Spares

Description

MME writes a spare-chip table to the mainframe that merges the spare-chip table configured in SCE with the bad bit codes that you specify with the Upper Bad Bit Code: ♥ and Lower Bad Bit Code: ♥ menus.

- 2. From the Upper Bad Bit Code: ♥ and Lower Bad Bit Code: ♥, choose the bad bit codes that you want to use. (These menus are available for the <u>merge with conliqued Spares</u> and <u>Overide Configured Spares</u> settings.)
- 3. Click on (Apply Selections); MME writes the spare-chip table to the mainframe.

Ensure that you reset the spare-chip table when you are done using a special spare-chip table configuration. You can reset the spare-chip table by clicking on Mode: <u>Meret-use Configured Sumer</u> and then clicking on <u>(Apply Selections</u>). You can also reset the spare-chip table by applying the configuration in SCE.

Viewing the Current Working Directory and Changing the Concurrent Maintenance Check and Debug Messages Settings

To view the current working directory and to change the concurrent maintenance check and debug messages settings, choose **Miscellaneous** from the Category: \bigtriangledown . The MME Resource Allocation window changes to:

© MME Resource Allocation	
Catagory: 🔽 Miscellaneous	
Current Working Directory: /tmp_mnt/data/nova/cmedev/t32	
Error Logger Access:	
Nonexclusivo Enclusive	
Concurrent Maintenance Check:	
Disabled Enabled	
Debug Messages:	
Channel Functions	
Utilities	
SCE & Reset	

Current Working Directory

The Current Working Directory field shows the directory that MME uses to access all relative paths [paths that do not start with a slash (/)] shown in MME windows. MME appends all relative paths to this directory to determine the absolute path that MME uses to access files. For example, if a window displays usr/seq and this field displays /cri/cme/t32, MME uses the /cri/cme/t32/usr/seq directory to access files. This field is read-only.

The information shown in this field is useful in the Systems Test and Checkout (STCO) environment in which technicians have access to remotely mounted directories of prereleased software. This information enables a technician to verify the paths used to load and save files so the technician can ensure that the proper software is being used.

Concurrent Maintenance Check

In environments 1 and 2, the Concurrent Maintenance Check setting specifies whether MME should check the concurrent mode (CONC) bit in the test code before loading a test when MME is in concurrent mode. Although this setting appears active in environment 0, it has no function in environment 0. Do not change this setting in environment 0: the Concurrent Maintenance Check setting should always be set to **Listuded**.

Future versions of MME will not include this setting in environment 0.

Debug Messages

The Debug Messages setting specifies the amount of output that MME returns to the standard output window.

Click on <u>Channel Functions</u> to have MME display the functions that MME sends to the mainframe (for example: the functions that MME uses to read mainframe memory).

Click on Utilities to have MME display status information about any utilities that it runs.

Click on <u>Set & Herset</u> to have MME display status information about any SCE functions or reset activities that occur.

Properties -> Enable Auto Apply



The Properties -> Enable Auto Apply command, as shown at the left, enables automatic application of function or utility changes in compose mode.

When the automatic apply function is enabled, you do not need to click on (Apply) in the MME Compose Sequence Entry window to apply any changes you make to a function or utility. Instead, move the cursor to the MME base window or to the menu bar in the MME Compose Sequence Entry window to automatically apply the changes.

Properties -> Disable Auto Apply



The Properties -> Disable Auto Apply command, as shown at the left, disables automatic application of function or utility changes in compose mode.

When the automatic apply function is disabled, you must click on (Apply) in the MME Compose Sequence Entry window to apply any changes you make to a function or utility.

Utilities -> Pattern



The Utilities -> Pattern command, as shown at the left, enables you to fill a selected portion of mainframe memory with a data pattern. This command displays the MME Memory Pattern Utility window:

S MME Memory Pattern Utility		
Mode:		
MWS All C		
C BAY Single		
Save & Restore Enviro	nment On Completion	
Pattern Select:		
Zeros	Even Bits	
Ones	Address	
Odd Bits	User Defined	
User Defined Format	b	
Byte Parkel	Halfword Word	
User Defined Patteri	3)	
Base: Absolute	Start: 00000000000	
Ctriot 8ase	Length: <u>00000000000</u>	
	Limit: 00000000000	
Sta	art	

Perform the following procedure to use the pattern utility:

1. Specify the pattern you want to use. The Pattern Select settings specify the following patterns:

Setting	Pattern
Zei os	The utility uses a pattern of all 0's.
Ones	The utility uses a pattern of all 1's.
Odd Bits	The utility uses a pattern that contains parcels of odd bits (125252 octal).
Even Bits	The utility uses a pattern that contains parcels of even bits (052525 octal).

Setting	Pattern
Address	The utility uses a pattern that contains the address of each memory location that is being patterned.
user Delined	The utility uses a user-specified pattern. Specify the format (click on User Defined Format: [byw], [bace], [Halfwood], or [wd]). In the User Defined Pattern field, enter the pattern you want to use.

- 2. Specify the area in mainframe memory that you want to write with the pattern. In the Start field, enter the first address that you want to write and then press the Return key. In the Length field, enter the length of the data block that you want to write and then press the Return key. In the Limit field, enter the last address that you want to write and then press the Return key.
 - **NOTE:** You must press the Return key after you enter data in these fields, or MME will not use the data. Entering data in two of the fields cause MME to automatically update the third field.
- 3. Click on <u>start</u>; MME uses maintenance channel functions to write the data pattern in mainframe memory.

Utilities -> Find

Utilities 🔻
Pattern
(Find
Copy/Move
Configuration
Logic Monitor

The Utilities -> Find command, as shown at the left, searches MME buffer memory for a data pattern. Use this command to locate all occurrences of a data pattern within a block of memory. This command displays the MME Find Utility window:

Q	MME Find Utility
Source:	
Search Boundary: Byte Parcel Halfword Word	
Pattern/Mask Size: Word	
Pattern/Mask Format: Byte Parcel Halfword Word	
Pattern: 000000 000000 000000 000000	
Mask: 177777 177777 177777 177777	
Address: 00000000000000000000000000000000000	
(Continue)	

The left side of the window contains the settings that specify the pattern to search for and the memory block to search. The right side contains a scroll box that displays the memory locations with a matching data pattern. The scroll box displays up to 256 entries; if more than 256 matches are found, the message 256 matches, additional occurrences known to exist is displayed in the lower-left corner of the window. To view the additional occurrences, click on the <u>Continue</u> button.

Perform the following procedure to manipulate this window.

- **NOTE:** Source defaults to **Bully** because the Utilities -> Find command in environment 0 searches MME buffer memory.
- 1. Specify the Search Boundary you want to use. The search boundary indicates the stride used for checking memory.

Click on by to check memory in byte increments, click on parel to check memory in parcel increments, click on mattered to check memory in halfword increments, or click on wide to check memory in word increments.

2. Specify the Pattern/Mask Size. The size indicates the size of the data pattern that is searched for and the mask that is used.

The settings available depend on the Search Boundary setting. Click on **by**, **page**, **mailwood**, or **word** to select the pattern and mask size.

3. Specify the Pattern/Mask Format. The format indicates the type of data pattern that is searched for and mask that is used.

The settings available depend on the Pattern/Mask Size setting. Click on **Prop**, **Pacel**, **Halfwod**, or **word** to select the pattern and mask format.

- 4. Specify the data pattern you want to search for in the Pattern field.
- 5. Specify the mask you want to use in the Mask field. The mask specifies which bits to compare. If a bit in the mask is set to 0, the bit position is not compared; if a bit in the mask is set to 1, the bit position is compared.
- 6. Specify the memory block you want to search (performing any two of the following actions automatically updates the third field):
 - Enter the first address of the memory block in the Address field, and press the Return key.
 - Enter the length of the memory block in the Length field, and press the Return key.
 - Enter the last address of the memory block in the Limit field, and press the Return key.
- 7. Click on <u>Start</u> to start the search. The <u>Start</u> button changes to <u>Stop</u>, and MME updates the MME Find Utility window.

Source:	0000000010000 000	000 000000 000000 000000	
Duffer			l l
Buffer			
Search Boundary:	00000000010003 000	000 000000 000000 000000	
search boundary:	00000000010004 000	000 000000 000000 000000	
Byte Parcel Halfword Word	00000000010005 000	000 000000 000000 000000	
	00000000010006 000	000 000000 000000 000000	
Pattern/Mask Size:	00000000010007 000	000 000000 000000 000000	
		000 000000 000000 000000	
Word		000 000000 000000 000000	
		000 000000 000000 000000	
Pattern/Mask Format:		000 000000 000000 000000	
Byte Parcel Halfword Word		000 000000 000000 000000	
Byte Farter Harword word		000 000000 000000 000000	
Pattern:		000 000000 000000 000000	
000000 000000 000000 000000		000 000000 000000 000000	
		000 000000 000000 000000 000 000000 000000	
Mask:			
177777 177777 177777 177777			
1///// 1///// 1///// 1/////		000 000000 000000 000000	
Address: 00000000010000			
Length: <u>00000000010000</u>			
Limit: 00000000020000			
Emme. 0000000020000			
(Stop) (Continue)			

Click on <u>Stop</u> to stop searching for the pattern; click on <u>Continue</u> to see the next set of entries if more than 256 occurrences exist.

Utilities -> Copy/Move

Utilities	
Pattern	
Find	
Copy/Move	
Configuration	
Logic Monitor	
	-

The Utilities -> Copy/Move command, as shown at the left, enables you to copy (or move) data in mainframe memory. This command displays the MME Copy/Move Utility window:

𝔇 MME Copy∕Move Utility			
Mode:	Mode:		
Parcel	Word		
Base:			
Absolute	Ctript 8ase		
Source:			
	0000000000000		
Length:	00000000000000000		
Limit:	: <u>000000000000</u>		
Destination: Start:	: 0000000000000000000000000000000000000		
July 1	<u></u>		
Сору	Move		

Perform the following procedure to manipulate this window:

- 1. Specify the Mode. Click on reaction to use parcel values for addresses. Click on reaction to use word values for addresses.
- 2. Specify the location of the source data. In the Start field, enter the first address of the block of data that you want to move or copy. In the Length field, enter the length of the block of data that you want to move or copy. In the Limit field, enter the last address of the block of data that you want to move or copy.

You must press the Return key after you enter data in these fields, or MME will not use the data. Entering data in two of the fields causes MME to automatically update the third field.

3. In the Start field, specify the destination where MME should write the copied/moved data.

You must press the Return key after you enter data in this field.

4. Click on Copy to copy the data, or click on Move to move the data. (When you move data, MME fills the source mainframe memory locations with 0's.)

Utilities -> Configuration



The Utilities -> Configuration command, as shown at the left, starts the System Configuration Environment (SCE). Use this application to configure the mainframe. For more information about SCE, refer to the *SCE User Guide*, publication number HDM-069-A.

Utilities -> Logic Monitor



The Utilities -> Configuration command, as shown at the left, starts the Logic Monitor Environment (LME). For more information about LME, refer to the *LME User Guide*, publication number HDM-070-0.

Reset -> Channel



The Reset -> Channel command, as shown at the left, resets the FY driver.

Reset -> Server



The Reset -> Server command, as shown at the left, resets the server. This halts any sequence(s) that MME is executing.

Reset -> Configuration



The Reset -> Configuration command, as shown at the left, causes SCE to reapply the configuration.

NOTE: This command does not work if any partitions have an OS owner. For more information about partition ownership, refer to the *SCE User Guide*, publication number HDM-069-A.

ENVIRONMENTS 1 AND 2

This section describes the interface components and menu button commands for MME environments 1 and 2.

Interface Components

MME environment 1 and 2 interfaces share a common interface. Use this interface to control testing in environment 1 and environment 2. Figure 3 shows the common interface. The paragraphs that follow the figure describe the components of the interface.

NOTE: For newer versions of MME, the interface contains only the number of CPUs that are available for your mainframe chassis type. For CRAY T94 mainframes, the interface contains 4 CPUs. For CRAY T916 mainframes, the interface contains 16 CPUs. For CRAY T932 mainframes, the interface contains 32 CPUs.

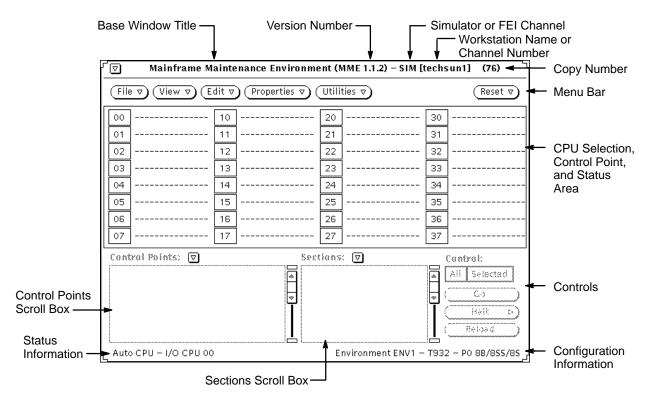


Figure 3. Environments 1 and 2 Interface Components

Base Window Title

The base window title displays the name of the program: Mainframe Maintenance Environment.

Version Number

The version number indicates the version of MME you are using.

Simulator or FEI Channel

The simulator or front-end interface (FEI) channel indicator shows MME is running with the simulator (indicated by SIM) or an FEI channel (indicated by FEI CHN 0 for channel 0, FEI CHN 1 for channel 1, FEI CHN 2 for channel 2, etc.).

Workstation Name or Channel Number

The workstation name or channel number indicator lists the name of the workstation or the channel number on which MME is running.

Copy Number

The copy number indicates the copy of MME you are using. To set the copy number, start MME with the -copy option. If you start MME with the default copy number of 0, the MME base window does not display a copy number. For more information about starting MME with the -copy option, refer to the *MME User Guide*, publication number HDM-102-A.

Menu Bar

The menu bar contains six menu buttons: $(File \nabla)$, $(View \nabla)$, $(Edit \nabla)$, $(Properties \nabla)$, $(Utilities \nabla)$, and $(Reset \nabla)$. For descriptions of the commands accessible from these menu buttons, refer to "Menu Button Commands" on page 60.

MME displays the following indicators in the menu bar when memory, register parity, shared register, LAT, and unknown errors are detected: MEM, RPE, SHR, LAT, and UKN. When you see any of these indicators, use the View -> Error Log menu button command to view the error log; refer to "View -> Error Log" on page 99 for more information about the error log. The following snap shows the MEM error indicator in the menu bar:

🕼 🛛 Mainframe Maintenance Environment (MME 1.0.7) – SIM [durand	i] (23) [\]
$(File \ \nabla) (View \ \nabla) (Edit \ \nabla) (Properties \ \nabla) (Utilities \ \nabla) (MEM)$	(Reset 🔻

CPU Selection, Control Point, and Status Area

The CPU selection, control point, and status area is where you assign CPU(s) to control points and where MME displays status information for running control points.

To assign CPUs to the current control point, click on any of the CPU settings (00 through the available number of CPUs in the mainframe). The CPU setting is highlighted and the control point name is displayed next to the CPU setting:

00) 00	csr.t
----	------	-------

MME displays several types of status information for the CPUs; to change the status information displayed, press the MENU mouse button in this area. The following menu appears:

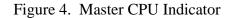
00	00 csr.	t [1:>]		30	
01		Active CPU Display	[_1]·	31	
02		Control Point	<u>[22</u>]·	32	
03		Pass Count (Global)	[<u>23</u>]·	🔝	
1.74		Pass Count (Section)	[24] -	34	
05		(P Register)	25	35	
-36-		CIP	[<u>28</u>] ·	38	
07 17		[<u>27</u>] ·	37		

The entries in this menu enable you to choose which status information is displayed.

Entry	Description
Control Point	Displays the control point name
Pass Count (Global)	Displays the pass count for all sections
Pass Count (Section)	Displays the pass count for the current section
P Register	Displays the current P register value
CIP	Displays the current instruction parcel (CIP) register value

The number shown to the left of the control point name is the control point copy number when several CPUs are assigned to a control point. A rectangle around the control point, as shown in Figure 4, indicates the master CPU for a group of CPUs assigned to a control point. The master CPU is the first CPU assigned to the control point.

00 00 srct.t 01 00 srct.t	02 00 srct.t	03
---------------------------	--------------	----



MME also automatically displays status information for executing control points. This information includes interrupt flags (in environments 1 and 2), which are indicated by IFLAG, as shown in Figure 5, and controller status messages (in environment 2 only), which are indicated by CSTAT, as shown in Figure 6).

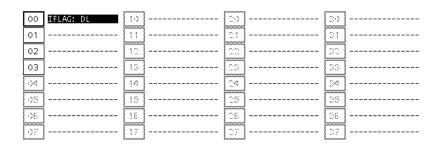


Figure 5. Example Interrupt Flag

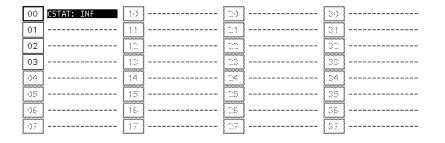


Figure 6. Example Controller Status Message

Table 1 describes the interrupt flags.

Interrupt Flag	Description
AMI	Address multiply range error interrupt
BPI	Breakpoint interrupt
DL	Deadlock
EEX	Error exit (000 issued)
FPE	Floating-point error
ICP	Internal CPU interrupt
IOI	Input/output (I/O) interrupt
MCU	MCU interrupt
MEC	Correctable memory error
MEU	Uncorrectable memory error
MII	001 <i>ij</i> does not equal zero or 033 instruction interrupt
NEX	Normal exit (004 issued)
PCI	Programmable-clock interrupt
PRE	Program range error
RPE	Register parity error
RTI	Real-time interrupt

Table 1. Interrupt Flags

Table 2 describes the controller status messages.

Status Message	Description
CIB	The control point attempted to reserve a cluster that is not in the data space.
CNR	The control point attempted to release a cluster that was not reserved.
CRE	A channel reservation error occurred: at least two CPUs were in the channel reservation code at the same time.
DMP	The control point dumped the CPU's registers and idled the CPU in the controller's idle loop.
HTM	The control point stopped and requested all CPUs to hang.
HTS	The control point stopped at the request of another CPU.
INF	The CPU exchanged to the controller with no interrupt flags.
IUC	An interrupt on an unreserved channel occurred.
LEBi	The logical base from dmpAREA is less than the starting logical base [for an exchange using IEXP (hDIFM)].
LBEI	The logical base from dmpAREA is less than the starting logical base [for a restart CPU load (cLOAD) function].
LBEx	The logical base from dmpAREA is less than the starting logical base [for an exchange using an XP table request (hXEXP)].
LES	The control point stopped on a logical address translation (LAT) table fault.
LLEi	The logical limit from dmpAREA is less than the starting logical limit [for an exchange using IEXP (hDIFM)].
LLEI	The logical limit from dmpAREA is less than the starting logical limit [for a restart CPU load (cLOAD) function].
LLEx	The logical limit from dmpAREA is less than the starting logical limit [for an exchange using an XP table request (hXEXP)].
MEI	The control point stopped on an invalid memory error.
MES	The control point stopped on a memory error (used MRSTOP).
MWS	MME sent a bad request.
OK	A normal operation occurred.
PEI	The control point stopped on an invalid parity error.
PES	The control point stopped on a parity error (used MRSTOP).
REQ	The CPU performed a dump and idle, but the mwsTOcpu request was not a dump request, or there was no idleSTAT and no mwsTOcpu request.
RES	MWS request was cleared before a register dump was completed.
SRE	A cluster reservation error occurred: at least two CPUs were in the cluster reservation code at the same time.
TRP	An invalid exchange occurred.
WTS	The control point stopped because of a dump and wait/resume request.
WTW	The control point is waiting because of a dump and wait/resume request.

Table 2. Control	er Status Messages
------------------	--------------------

Status Message	Description
XNRi	An exchange occurred using a cluster that was not reserved [for an exchange using IEXP (hDIFM)].
XNRI	An exchange occurred using a cluster that was not reserved [for a restart CPU load (cLOAD) function].
XNRx	An exchange occurred using a cluster that was not reserved [for an exchange using an XP table request (hXEXP)].

Table 2. Controller Status Messages (continued)

Controls

The controls are buttons and settings that enable you to start or stop running control points and reload control points.

Button/Setting	Description
All	Performs the control button command on all CPUs that have been assigned to control points.
Selected	Performs the control button command on the selected control points.
	Starts the control points.
(Halt ▷)	Stops the control points. Three options are available:
	Halt -> No Dump sets Master Clear on the CPU(s). This halt option is not available for environment 2.
	Halt -> Exchange Dump sets Master Clear on the CPU(s) and uses a maintenance channel feature to perform an exchange dump. This halt option is not available for environment 2.
	Halt -> Register Dump sets Master Clear on the CPU(s), builds a starting exchange package (SEXP) that points to dump memory, and restarts the CPU(s). The CPU(s) then dump the registers at memory location dmpAREA.

Button/Setting	Description
(continued)	For detailed information about what happens when you choose one of the halt options, refer to the <i>MME User Guide</i> , publication number HDM-102-A.
Reload	Reloads the control points. This removes all user changes for the control points. (User changes load when you click on) or when a new section starts to execute.)
Resume	Resumes execution of a holding control point. This button appears when a control point hold request executes.

Configuration Information

The configuration information displays the current MME environment (environment 1 or 2) and the current configuration data (mainframe type; partition where MME is running; and number of sections, subsections, and banks). The System Configuration Environment (SCE) provides this information.

For example, the following configuration information indicates that MME is in environment 1 (Environment ENV1); the mainframe type is a CRAY T94 system (T94); MME is running in partition 0 (P0); and the configuration consists of 8 banks (8B), 1 subsection (1SS), and 8 sections (8S):

Environment ENV1 – T94 – PO 8B/1SS/8S

The configuration information component displays the following mainframe types:

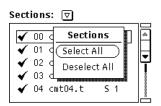
Type	Description
Tester	A tester with 1 CPU and 1 memory module
T94	A CRAY T94 mainframe
T916	A CRAY T916 mainframe
T932	A CRAY T932 mainframe

Sections Scroll Box

The Sections scroll box displays the sections that are available to run for the current control point. This scroll box shows the section number; the file that contains the section; a code that indicates whether the section is a single-CPU (S), multiple-CPU (M), or single-CPU or multiple-CPU (M/S) section; and numbers that indicate whether the section can be run in environment 1 (1), environment 2 (2), or environments 1 and 2 (1/2).

NOTE: Sections that run only in multiple CPUs include a number that indicates the minimum number of CPUs required to use the section. If you assign fewer than the required number of CPUs to the control point section, MME automatically deselects the section. For example, if the Sections scroll box displays M02, this indicates that the section requires at least 2 CPUs.

When you select a control point in the Control Points scroll box, the Sections scroll box displays the test sections. Sections that are selected to run when () is clicked have a check mark (\checkmark) next to them. To select or deselect a section, move the mouse pointer over the section and click the ADJUST mouse button. To select or deselect all sections, move the mouse pointer over the Sections scroll box, and press the MENU mouse button. The following menu appears:



NOTE: The Sections: menu also contains these options.

Choose **Select All** to select all sections of the test; choose **Deselect All** to deselect all sections of the test.

To select or deselect sections for testing while control points are running, use the ADJUST (for single sections) or MENU (for all sections) mouse buttons.

To choose the current section, move the mouse pointer over the section and click the SELECT mouse button; a box surrounds the section to indicate that it is the current section, and the section is loaded into mainframe memory. User changes to the section also load when you choose a new current section. (User-loaded data does not load unless you previously saved the control point in <u>Current Section (Memory Integr</u>) mode; refer to "File -> Save -> Control Point" on page 69 for more information.)

Status Information

The status information displays the current state of the MME program by using the following messages:

Message	Description
Auto CPU	Enables automatic assignment of control points. When a control point is loaded, MME assigns a system-selected CPU to it. For more information about changing this option, refer to the "Properties –> Resource Allocation" description later in this section.
I/O CPU ##	The CPU specified by ## is the I/O CPU. (The I/O CPU specifies which CPU path MME uses to access mainframe memory.)
I/O Disabled	No CPU is set as the I/O CPU.

Control Points Scroll Box

The Control Points scroll box shows the current control point loaded in MME. If you press the MENU mouse button in this scroll box, the following menu appears:

Control Points: 🔽			
✔ 00	Control Poir	n ts g.cp02.4/	
	View By	Filename	F
		Location	T
		A	

NOTE: The Control Points: \bigtriangledown menu also contains these options.

Choose **Select All** to select all control points. Choose **Deselect All** to deselect all control points. Choose **View By -> Filename** to see the control point indicated by its filename:

Control Points	. 🗸	
✓ 00 cmt.1	rel/diag.cp02.4/	

Choose **View By -> Location** to see the control point listed by its location in mainframe memory:

Cont	rol Points:	\bigtriangledown	
✓	00 cmt.t	000000000000000000000000000000000000000	*
			ľ

When a control point is executing in a CPU, the Control Points scroll box displays the global pass count for the control point, as shown in the following snap:

Control Points:	\bigtriangledown	
✔ 00 cmt.t	0000000000] 🖗
		P

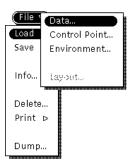
If a control point detects an error, the Control Points scroll box displays ERR next to the global pass count for the control point, as shown in the following snap:

Control Points:	\Box		
✔ 00 cmt.t	00000000000	ERR	

Menu Button Commands

The menu buttons contain commands you use to manipulate MME environments 1 and 2. This subsection describes what each command does and how to use each menu button command. Figure 10 on page 155 shows all available menu button commands for environment 1. Figure 11 on page 157 shows all available menu button commands for environment 2.

File -> Load -> Data



The File -> Load -> Data command, as shown at the left, loads a data set into mainframe memory. Use it to re-create a specific data set for testing without having to manually enter it each time. This command displays the MME Load/Save Data window:

Ø MM	E Load/Save Data
Mode:	Word Parcel
Load Base:	Absolute Relative
Load Directory:	⊽ usr/data/*
Load Files:	bg bg1 bg2 bob
Load Type:	
	00000000000000
Save Base:	Absolute Relative
Save Directory:	usr/data
Save File:	
	000000000000000000000000000000000000000
Save Length:	00000000000000000000000000000000000000
	78 files found

Perform the following procedure to manipulate this window:

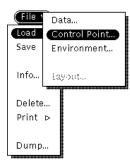
- 1. Select the data Mode. Click on word to use a word load address and length or word to use a parcel load address and length. The Load Address and Load Length fields change to the specified format.
- 2. Specify the Load Base. Click on *Ausoluse* to load the data at a fixed location in memory or *Melative* to load the data relative to the base location in memory of the current control point section.

3. Change the directory, if necessary, by choosing a directory from the Load Directory: row or by entering the directory in the Load Directory field.

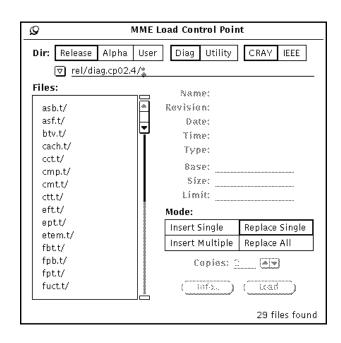
The Load Directory: includes selections for the usr diagnostic and data directories, all of the rel directories, and all of the alpha directories.

- 4. In the Load Files scroll box, click on the data file you want to load. The Load Type entry displays what type of data is contained in the file. The Load Length field displays the length of the chosen data file.
- 5. In the Load Address field, enter the address at which the data should be loaded. The Load button activates.
- 6. Click on (Load); MME loads the data at the specified address.
- **NOTE:** A current software bug overwrites any user-loaded data when you click on ______. To use user-loaded data, perform the following procedure, which saves the data with the control point section:
 - 1. Load the control point. Make the section you want to use the current section.
 - 2. Load the data with the File \rightarrow Load \rightarrow Data command.
 - Save an image of the current control point section. (Use the Current Section (memory image) mode to save the section; refer to
 "File -> Save -> Control Point" on page 69 for more
 information.) This stores the data with the control point
 section.

File -> Load -> Control Point



The File -> Load -> Control Point command, as shown at the left, loads a control point to use for testing. Use it to access control points from the current offline diagnostic release, to access prereleased versions of the control points, and to access user-created or modified control points. This command displays the MME Load Control Point window:



This window displays the control points you can load under MME. The directory path displayed to the right of the Dir: 🕤 indicates the current directory. All control points are divided into sections, which are files contained in subdirectories of the current directory. The Files scroll box displays the subdirectories that hold the control point sections. When you load a control point, the Sections scroll box in the MME base window displays the sections.

Perform the following procedure to manipulate this window:

- 1. Change the directory, if necessary, by performing one of the following actions:
 - Click on the appropriate Dir settings:

Setting	Description
Aclase	Selects the directory that contains files from the current offline diagnostic release
Alpha	Selects the directory that contains prereleased files
User	Selects the directory that contains files you have saved or modified
Diag	Specifies that you want to load a diagnostic program
utility	Specifies that you want to load a diagnostic utility
CRAY	Specifies that you want to use diagnostic test programs or utilities written for Cray Research, Inc. (CRI) floating-point CPUs
IEEE	Specifies that you want to use diagnostic test programs or utilities written for IEEE floating-point CPUs

• Choose a directory from the Dir: ♥ or enter the directory in the Dir field. The following directories are available in Dir: ♥:

The Dir: v includes selections for the usr diagnostic directory, all of the rel directories, and all of the alpha directories. The RESET CHOICE DEFAULTS selection resets the field to the directory you selected with the Mekawa, Multa, User, User, User, User, CPAN, and IEEE settings.

NOTE: A .t extension indicates a control point assembled in Triton mode. A .c extension indicates a control point assembled in C90 mode. A .e extension indicates a control point assembled for IEEE CPUs.

2. Click on the control point you want to load. The following description information is updated:

Field	Description	
Name	Name of the control point	
Revision	Revision level of the control point	
Date	Date the control point was assembled	
Time	Time the control point was assembled	
Туре	Diagnostic, Utility, or Loop indicator	
Base	Address where the control point is loaded	
Size	Octal size of the control point code that is loaded into memory	
Limit	Last available address for the control point	

- 3. Change any or all of the Base, Size, and Limit values, if necessary. When you change any two of these values, this window updates the third automatically.
 - Base indicates the address in memory where the control point is loaded. Normally, this should be zero.
 - Size indicates the size of the control point. Normally, this should be the default value. (Press the Return key after you change this value.)
 - Limit indicates the maximum address for the control point. Normally, this should be the default value. (Press the Return key after you change this value.)

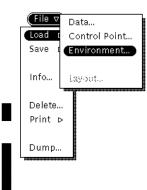
4. Change the Mode setting to specify how MME loads the control point. Click on one of the following settings:

Setting	Description
tusert Single	Loads a new control point.
	If a control point is already loaded in environment 1, an error message appears that indicates no memory is available because usually only one control point can be loaded at a time in environment 1.
	If a control point is already loaded in environment 2, the new control point is also loaded if enough memory is available.
Insert Multiple	Loads multiple copies of the control point you specify. This option is for environment 2 only. In the Copies field, specify the number of copies to load.
Replace Single	Replaces a loaded control point with the control point you specify; use this option to replace a control point that is already loaded.
Replace All	Replaces all loaded control points with the control point you specify This option is for environment 2 only.
Click on Load ;]	MME loads the specified control point.
TE: The <u>Info</u> but	utton displays the information file for the control

NOTE: The <u>Info...</u> button displays the information file for the control point in the MME Control Point Information window. Refer to the "File -> Info" description on page 72 for more information about this window.

5.

File -> Load -> Environment



The File -> Load -> Environment command, as shown at the left, enables you to load environment data that you previously saved with the File -> Save -> Environment command. Use this command to load the following information:

- The environment that you were using (environment 1 or 2)
- The control points that were loaded
- The changes that you made to those control points
- The CPU assignments for the control points
- The saved resource and memory allocation settings

This command displays the MME Load/Save Environment window:

🖉 🛛 MME Load/Save Environment
Load Dir: usr/env/*
mtb/ testenv/ testenv2/
Save Dir: usr/env Save File:
Save

Perform the following procedure to manipulate this window:

- 1. In the Load Dir field, enter the directory in which the environment data was saved.
- 2. In the Files scroll box, click on the file you want to load.
- 3. Click on <u>Load</u>; MME loads the environment data in the specified file.

File -> Load -> Layout

This feature has not been implemented yet.

File -> Save -> Data



The File -> Save -> Data command, as shown at the left, saves a data set so you can reuse it. Use it to re-create a specific data set for testing. This command displays the MME Load/Save Data window:

Ф мм	MME Load/Save Data		
Mode:	Word Parcel		
Load Base:	Absolute Relative		
Load Directory:	⊽ usr/data/*		
Load Files:	bg bg1 bg2 bob		
Load Type:	Plain File		
	000000000000000000000000000000000000000		
Load Length:	occoccoccocco (Lead)		
Save Base:	Absolute Relative		
Save Directory:	usr/data		
Save File:			
	000000000000000000000000000000000000000		
Save Length:	00000000000000000000000000000000000000		
	78 files found		

Perform the following procedure to manipulate this window:

- 1. Select the data Mode. Click on wide to use a word save address and length or wide to use a parcel save address and length. The Save Address and Save Length fields change to the specified format.
- 2. Specify the Save Base. Click on Ausolute to save the data from an absolute memory location (based on a starting address of 0), or click on Automative to save the data relative to the base address of the current control point section.
- 3. Specify a different directory in the Save Directory field, if necessary.

4. Specify the name of the file in the Save File field.

Pressing the Return key in the Save File field saves the data file (the same action as when you click on the (Save) button). Before you press the Return key in this field, update the Length field with the correct value, or MME will save a zero-length file.

- 5. Specify the starting address of the data block you want to save in the Save Address field. (Press the Return key after you change this value.)
- 6. Specify the length of the data block you want to save in the Save Length field. (Press the Return key after you change this value.)
- 7. Click on (Save); MME saves the specified data set.

File -> Save -> Control Point



The File -> Save -> Control Point command, as shown at the left, saves a control point you have modified. Use it to create customized control points for troubleshooting. This command displays the MME Save Control Point window:

9	MME Save Control Point		
Mode:	All Sections (User Changes Only)		
	Current Section (Memory Image)		
Dire	e ctory: usr/diag		
	File: newcp		
	Name: fpt.t Copy: 00		
	vision: TRI 4.1		
	Type: Diagnostic 📋 Sava As Lo	x·p	
Memory	Memory MAX: 🗹 All Available		
Sectio	Section: 001		
01	Origin: Length: 00000015100		
03	Min Pass: 00000000001		
04	Max Pass: 37777777777		
05	Min Pass: 00000000001 Max Pass: 377777777777777777777777777777777777		

Perform the following procedure to manipulate this window:

1. Specify the Mode. Click on <u>All Sections Toter Changes Confi</u> to save the changes you have made; this creates a new information file that contains the user changes for the control point (minimum and maximum memory) and all sections (minimum and maximum pass counts) and makes hard links to the original files for the sections.

Click on <u>Current Section (memory Image</u>) to save a memory image of the current control point section; this creates an information file that contains the minimum and maximum memory and minimum and maximum pass counts for the section. You may adjust the length of the memory image to save data outside the normal range for the section; this is useful for capturing diagnostic-created data (for example, a diagnostic-generated instruction block).

2. In the Directory field, enter the name of the directory in which you want to save the control point.

- 3. In the File field, type the name of the file in which you want to save the control point; the Name field shows the current name of the control point, the Revision shows the revision of the offline diagnostic, and the Type field indicates the type of file.
- **NOTE:** To save a diagnostic as a loop so MME will not configure the diagnostic when loading it, click on the Save As Loop check box.
- 4. In the Memory MIN field, specify the minimum amount of memory the control point can be assigned. If the specified amount of memory is not available, the control point will not load when you use the File -> Load -> Control Point command.
- 5. In the Memory MAX field, specify the maximum amount of memory the control point requires, or click in the check box next to All Available to specify that the control point should use all available memory.
- 6. If you are saving a loop, specify, in the Origin field, the starting address of the loop in mainframe memory.
- 7. If you clicked on <u>Curant Section (Memory Image</u>), specify the length of the memory image you want to save in the Length field.
- 8. If you clicked on All Sections Toter Changes Confer, click on the section, in the Sections scroll box, for which you want to change the minimum and maximum pass counts. Repeat Steps 8 through 10 for each section you want to modify.
- 9. In the Min Pass field, specify the minimum number of passes the selected section must complete before MME can load a different control point section.
- 10. In the Max Pass field, specify the maximum number of passes the section should complete before MME disables testing of the section.
- 11. Click on <u>Save</u>; MME saves the modified control point information.

File -> Save -> Environment

File 🗸		
Load	Data	
Save	Control Point	
Info	Environment	
Delete		
Print		
Dump		

The File -> Save -> Environment command saves the current environment data so you can restore MME to its current state. Use this command to save the following information:

- The environment that you are using (environment 1 or 2)
- The control points that are loaded
- The changes that you have made to these control points
- The CPU assignments for the control points
- The current resource and memory allocation settings

This command displays the MME Load/Save Environment window:

🖉 MME Load/Save Environment
Load Dir: usr/env/* Load Files:
mtb/ testenv/ testenv2/
Save Dir: <u>usr/env</u> Save File:
Save

Perform the following procedure to manipulate this window:

- 1. In the Save Dir field, enter the directory in which you want to save the environment data. Usually, you should enter **usr/env** in this field.
- 2. In the Save File field, enter the filename under which you want to save the environment data.
- 3. Click on <u>Save</u>; MME saves the environment data in the specified file.

File -> Save -> Layout

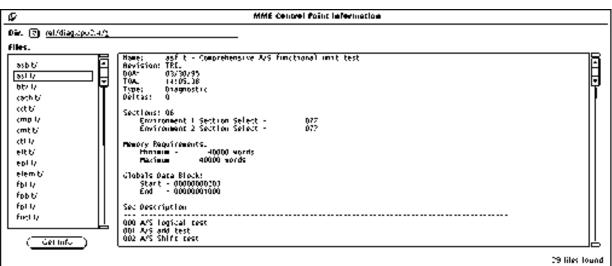
This feature has not been implemented yet.

File -> Info



The File -> Info command, as shown at the left, displays information about control points. Use this command to view information that includes the section names, descriptions, paths, and resource requirements.

This command displays the MME Control Point Information window:



NOTE: The <u>Info...</u> button in the MME Load Control Point window also displays the MME Control Point Information window.

Perform the following procedure to manipulate this window:

- 1. Change the directory from which MME should get the control point information, if necessary, by:
 - Entering the directory in the Dir field and pressing the Return key, or
 - Choosing the directory from the Dir: $\overline{\bigtriangledown}$.

- 2. In the Files scroll box, click on the control point for which you want information.
- 3. Click on <u>Get Info</u>. The window displays all available information from the control point.

File -> Delete



The File -> Delete command, as shown at the left, deletes file(s) you no longer need. Use this command to delete unwanted files from the MME user directories stored on the maintenance workstation (MWS). This command displays the MME Delete File window:

S MME Delete File	
Dir: ⊽ usr/diag/*	
Files:	
addrp/ ept.tmp/ find/ forcemec/ fpblp/ fpblp1/ fpblp2/ fpblp3/ fpmbg/	
fpblp1/ fpblp2/ fpblp3/ fpmbg/ fpmbg1/ fpmbg2/ fpmbg3/ fprbg/ fpt3lp/	
() 39 files found	

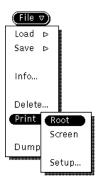
Perform the following procedure to manipulate this window:

- 1. Change the directory, if necessary, by:
 - Entering the directory in the Dir. v field and pressing the Return key, or
 - Choosing the directory from the Dir. v button. The following user directories are available.

Directory	Description
usr/data usr/diag usr/env usr/layout usr/lst usr/seq	User data files User diagnostic programs User environment files User layout data User listings User sequences
·	1

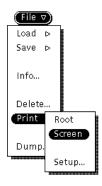
- 2. Click on the file you want to delete.
- 3. Click on **Later**). MME deletes the file.

File -> Print -> Root



The File -> Print -> Root command, as shown at the left, prints an image of everything contained in the root window, including the MME base window.

File -> Print -> Screen

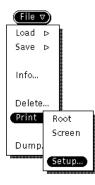


The File -> Print -> Screen command, as shown at the left, prints an image of a window or an icon.

When you choose this command, the cursor becomes a plus symbol (+). Move the cursor to the window or icon that you want to print, and click any mouse button.

NOTE: This command does not print an image of the MME base window. To print an image of the MME base window, use the File -> Print -> Root command.

File -> Print -> Setup



The File -> Print -> Setup command, as shown at the left, enables you to edit the commands that control how MME prints data for the File -> Print -> Root and File -> Print -> Screen commands. This command displays the MME Print Setup window:

MME Print Setup			
Print Roo	ot Command: 💽 root xpr –scale 3 –device ps –gray 3 /bin/lp)8		
Print Scree	int Screen Command: 💽 me xpr -scale 3 -device ps -gray 3 /bin/lp)&		
Print Te:	Print Text Command: Ip		
(<u>Sava</u> (Commands To File		

Modify the commands in the Print Root Command and Print Screen Command fields to change how MME prints. For more information about the UNIX xwd, xpr, and 1p commands used in the print processes, refer to the UNIX online manual (man) pages (enter **man xwd**, **man xpr**, or **man 1p** at a UNIX command prompt).

Use the buttons in the MME Print Setup window to perform the following functions:

- To save the current printer setup commands for later use, click on Save Commands To File.
- To load the printer setup commands that you saved previously, click on Reset Commands From File.
- To load the default printer setup commands that the MME program provides, click on (Reset Commands From Defaults).

File -> Dump



The File -> Dump command, as shown at the left, sends a copy of mainframe data to a file or the printer. Use this command to create a permanent record of the data so you can analyze it later. This command displays the MME Dump Setup window:

Q	MME Dump	Setup	
Mode:			
File	Printer C	ompress	
Director	nyo <u>usid</u>		
	File:		
Forma	at: 🔽 Parcel		
Bas	se: Absolute	Relative	
Start Address: 00000000000			
-	Length (words): <u>00000000000</u> End Address: 00000000000		
Comments:			
* *			
Dump			

Perform the following procedure to manipulate this window:

1. Specify the Mode. Click on **File** to output the data to a file, or click on **Prime** to output the data to the printer.

Click on <u>Compress</u> to compress the output of the File -> Dump command. This reduces the size of the data listing by replacing repeated lines with a statement similar to Last line repeated 077 (63) times.

2. If data is being dumped to a file, specify the directory in the Directory field and the file in the File field.

3. Choose the display format of the data from the Format: ♥. The following formats are available:

Format	Description
Nibble	Nibble data
Byte	Byte data
Parcel	Parcel data
Halfword	Halfword data
Word	Word data
Hexadecimal	Hexadecimal data
Address	Address data
Text	Text data
Exchange (NI)	Exchange data (not implemented)
Instruction (T90)	Instruction data (CRAY T90 series instructions)
Instruction (IEEE)	Instruction data (IEEE instructions)
Instruction (C90)	Instruction data (CRAY C90 series instructions)

- 4. Specify the Base. Click on *Ausolute* to dump data from an absolute memory location (based on a starting address of 0) or *Autometer* to dump data relative to the base address of the current control point section.
- 5. Specify the data block you want to dump by completing any two of the following actions (the third field is automatically set using data from the other two fields):
 - In the Start Address field, enter the starting address of the data block to dump and press the Return key.
 - In the Length (words) field, enter the length (in words) of the data block and press the Return key.
 - In the End Address field, enter the ending address of the data block you want to dump and press the Return key.
- 6. Click in the Comments scroll box and type any comments you want to include with the data. For example, you might specify that the data is from a specific test or that you created it on a specific date.
- 7. Click on <u>Dump</u>; MME sends the specified data to the printer or file.

View -> Memory

View 🗸	
Memory	
CPUs	⊳
Register Dump	
Error Log	
Memory Map	
Runtime Information	⊳
Listing	⊳
Notes	

The View -> Memory command, as shown at the left, displays mainframe memory in a separate window. Use this command to verify the contents of specific memory locations or to change the data stored in memory. This command displays the MME View Memory Setup window:

Ø MM	🖉 👘 MME View Memory Setup						
	Refresh (msec): 1000 🛛 🗕 🗁 🛛						
Mode:							
Memory	Exchai	nge	Instruction				
Format:							
Nibble	Halfv	vord	Text				
Byte	Word		Address				
Parcel	Hex						
Base:			_				
Absolu	ute		Relative				
Driftir	ng		Anchored				
_	Size:						
Address: Q							
View							

Perform the following procedure to manipulate this window:

- 1. To set the interval at which memory windows are updated, move the Refresh (msec) slider or enter a new value in the Refresh (msec) field. (Setting this value too low can monopolize the workstation CPU and reduce system performance.)
- 2. Click on a Mode (<u>Neiner</u>, <u>Ecchange</u>, or <u>Instruction</u>) to specify the way you want the data displayed.
- Click on a Format. The following Format settings are available for mode: <u>Nibble</u>, <u>Halfword</u>, <u>Text</u>, <u>Byre</u>, <u>wird</u>, <u>Address</u>, <u>Parcel</u>, and <u>Hes</u> (hexadecimal).

The following Format settings are available for Exchange mode: [Byte], [Word], [Pace], [Text], [Halfword], and [Address].

The following Format settings are available for <u>Instruction</u> mode: (Su), TSU), and IEEE. Memory mode (<u>Namay</u>) displays normal memory:

Q	Memo	ry – Ab	solute	
000000000000	000000	000000	000000	000000
00000000000000001	000000	000000	000000	000000
0000000000000002	000000	000000	000000	000000
000000000003	000000	000000	000000	000000
0000000000000	000000	000000	000000	000000
000000000005	000000	000000	000000	000000
000000000006	000000	000000	000000	000000
000000000007	000000	000000	000000	000000
000000000010	000000	000000	000000	000000
00000000011	000000	000000	000000	000000
00000000012	000000	000000	000000	000000
000000000013	000000	000000	000000	000000
000000000014	000000	000000	000000	000000
000000000015	000000	000000	000000	000000
000000000016	000000	000000	000000	000000
000000000017	000000	000000	000000	000000
000000000020	000000	000000	000000	000000
000000000021	000000	000000	000000	000000
000000000022	000000	000000	000000	000000
000000000023	000000	000000	000000	000000
000000000024	000000	000000	000000	000000
00000000025	000000	000000	000000	000000
00000000026	000000	000000	000000	000000
00000000027	000000	000000	000000	000000
00000000030	000000	000000	000000	000000
00000000031	000000	000000	000000	000000
00000000032	000000	000000	000000	000000
00000000033	000000	000000	000000	000000
00000000034	000000	000000	000000	000000
00000000035	000000	000000	000000	000000
00000000036	000000	000000	000000	000000
00000000037	000000	000000	000000	000000

Exchange mode (techange) displays exchange information:

Q	Memory – Absolute	
ADX P PN XA EA0 EA1 EA2 EA3 EA4	0000000000000 000000	7 10 10 10
CN 0	D00 MODES 015 BR RST EBK IM 00000000 II III FII III III IFI III II 200 DM MCR SDK RU FOP EBC MRI IPD MNA XNU OD D1 DEL LM PM PRR XPM CTP OCL IXM IXX VV	N N N
LATO LAT1 LAT2 LAT3 LAT4 LAT5 LAT6 LAT7	D RWWC 02 RWWD 02 PB 000000000000 LB 000000000000 LL 17777777740000 1 RWXC 00 RWXD 00 PB 000000000000 LB 0000000000000 LL 00000000	-

MME displays the same data for CRI floating-point format exchange packages and IEEE floating-point format exchange packages. The Memory window contains information for both formats. MME highlights only the bits and modes that are used for the CP modules included in the mainframe. This document shows exchange packages for CP modules that use the CRI floating-point format.

For more information about the contents of CRI floating-point format exchange packages, refer to the *CPU Module (CP02)* document, publication number HTM-003-0. For more information about the contents of IEEE floating-point format exchange packages, refer to the *IEEE CPU Module (CP02)* document.

Instruction mode (

Q			Memory (0) [T90]	
0000011000a	100100 000210		A1	210,0
0000011000d	120300 000212	000000	53	212,0
0000011001c	042277		52	1
0000011001d	056207		52	S2 <a7< th=""></a7<>
0000011002a	045332		53	#52&53
0000011002b	031110		A1	A1-1
0000011002c	042277		52	1
0000011002d	056201		52	S2 <a1< th=""></a1<>
0000011003a	044032		<u>50</u>	\$38\$2
0000011003b	014000 044022	000000	JSZ	11004c
0000011004a	045332		53	#S2&S3
0000011004b	001411		SIPI	A1
0000011004c	031110 030001		A1 A0	A1-1
		000000		A1
0000011005a 0000011005d	012000 044012 006000 040400		J AP J	11002c 10100a
0000011005u	000000 040400	000000	ERR	TUTUUA
0000011006C	000000		ERR	
00000110080	000000		ERR	
0000011007b	000000		ERR	
0000011007c	000000		ERR	
0000011007d	000000		ERR	
0000011010a	040100 001000	000000	S1	1000
0000011010d	044031	000000	so	\$3&\$1
0000011011a	042177		51	1
00000110116	044331		53	\$3851
0000011011c	014000 044105	000000	ĴŚŹ	11021b
0000011012b	120100 023200		51	23200,0
0000011013a	051003		ŝo	53
0000011013b	015000 044064	000000	JSN	11015a
0000011014a	045212		52	#S2&S1
0000011014b	06000 044074	000000	J	11017a

4. Specify a Base. Click on to display absolute memory addresses, based on a starting address of 0. This is the setting you will normally use in environment 1 because only one control point is loaded.

Click on <u>metative</u> to display memory addresses that are relative to the base address of the current control point. The window displays different areas of memory, depending on the base address of the current control point. This setting is useful for looking at the contents of multiple control points in environment 2.

Click on <u>triffing</u> to display memory locations for the current control point as you change control points. The window "drifts" to the base address for the current control point as you change control points.

Click on <u>Muchaned</u> to always display memory for the control point that was current when the memory window was first displayed. The window becomes "anchored" to the base address for one control point and always displays memory for that control point.

5. Choose the size of the display window from the Size: . This affects only the memory and instruction mode windows.

The following window sizes are available:

Size	Description
Small Medium	The window displays 10_8 words. The window displays 20_8 words.
Large	The window displays 40_8 words.
X–Large	The window displays 100_8 words.

6. Choose the font size you want to display in the window from the Font: ☑.

The following font sizes are available:

Size	Description
Small	The font size is small.
Medium	The font size is medium.
Large	The font size is large.
X–Large	The font size is extra large.

- 7. Change the starting address, if necessary, by double clicking on the Address field and typing a new value.
- 8. Click on <u>uiew.</u>. MME displays a window for the specified memory location.

If you want to change the Format, Memory, Exchange, Instruction, Window Size, or Window Font from the Memory – Absolute window, press the MENU mouse button and choose the menu item.

Q		Memo	ry – Ab	solute	
00000000	າດດດ			000000	000000
00000000	1001		000000	000	0000000
00000000	Form	at		⊳ ∎000	000000
00000000				500	000000
00000000				500	000000
10000000	Mem	ory (Met	ta-M)	500	000000
10000000		•		_ 500	000000
0000000	Excha	ange (Mi	eta-X)	D 00	000000
0000000	Instru	uction (I	Meta-I)	ь В ОО	000000
0000000				٥00 D	000000
0000000				D00	000000
0000000	Wind	ow Size		⊳ ₽ 00	000000
0000000				D00	000000
0000000		ow Font	-	Þ ∎00	000000
0000000th					000000
00000000		000000	000000	000000	000000
00000000		000000	000000	000000	000000
00000000		000000	000000	000000	000000
00000000		000000	000000	000000	000000
00000000		000000	000000	000000	000000
00000000		000000	000000	000000	000000
00000000		000000	000000	000000	000000
00000000		000000	000000	000000	000000
00000000		000000	000000	000000	000000
00000000		000000	000000	000000	000000
		000000	000000	000000	000000
		000000	000000	000000	000000
		000000	000000	000000	000000
		000000	0000000	000000	
000000000000000000000000000000000000000		000000	0000000	000000	0000000
		000000	000000	0000000	0000000
	5037	000000	000000	000000	000000

For example, the following window appears if you choose the Exchange (Meta–X) format menu option:

Q							Memor	y - Abs	olu	te				
P PN XA EA0 EA1	0000004 0000041 000 0000140 0000140 0000140 0000140 0000140	1000a))))	A0 01 A1 01 A2 01 A3 01 A3 01 A5 01 A5 01	00000 00000 00000 00000 00000 00000 0000)000)000)000)000)000)000)000		000100 000100 000000 000000 000000 000000	50 51 52 53 54 55 56 57	000000 177777 000000 000000 000000 000000 000000	000000 177777 000000 000000 000000 000000 000000	000000 177777 000000 000000 000000 000000 000000	1773 0001 0001 0001 0001	777 000 000 000 000 000 000
CN C VL 2	000 200	MODES	015	DM N	icr s	BM DM M	IM (0000000	II RU PM	1.01 66	Î MRÎ Î	II IFI PD MNA CL IXM	XNU (III ODN VVV
STAT		(NU ODI EXN VV SSS SS	V T	BB	**V **N **U	FWB PSM S L	IF (0000000	RM PE EU	FOP EBI PRR EPI EEE XI	E CTC O	CL IEM	IXN '	ODN VIV FVI
LATO LAT1 LAT2 LAT3 LAT4 LAT5 LAT6 LAT7	I RWXC (2 RWXC (3 RWXC (4 RWXC (5 RWXC (5 RWXC (5 RWXC (02 RW 00 RWXI 00 RWXI 00 RWXI 00 RWXI 00 RWXI 00 RWXI 00 RWXI	D 00 D 00 D 00 D 00 D 00 D 00 D 00	PB 0 PB 0 PB 0 PB 0 PB 0 PB 0 PB 0			00000 00000 00000 00000 00000 00000	B 00000 B 00000 B 00000 B 00000 B 00000 B 00000 B 00000		000000 000000 000000 000000 000000 00000	LL 0000 LL 0000 LL 0000 LL 0000 LL 0000 LL 0000 LL 0000	7777740 0000000 0000000 0000000 0000000 000000		

You can also change the window format (Format), data type (Memory or Instruction), window size (Window Size), or window font (Window Font) from this menu.

Instead of using the MENU mouse button, you may also use the diamond-shaped meta key (or , depending on the type of keyboard you have) with one of several keyboard shortcuts. The following key combinations are shortcuts to the menu options:

Key Sequence	Function
Meta-a	Switches the display to address format
Meta-n	Switches the display to nibble format
Meta-b	Switches the display to byte format
Meta-p	Switches the display to parcel format
Meta-h	Switches the display to halfword format
Meta-w	Switches the display to word format
Meta-e	Switches the display to hexadecimal format
Meta-t	Switches the display to text format
Meta-i	Switches the display to instruction mode
Meta-x	Switches the display to exchange mode
Meta-m	Switches the display to memory mode

Changing Memory

Perform the following procedure from a memory window to change data stored in mainframe memory:

1. Use the arrow keys to move the cursor to the location in memory you want to change, or click on the location. In this example, parcel 0000000005b was selected:

Q	Memo	ry – Ab	solute	
000000000000000000000000000000000000000	000000	000000	000000	000000
000000000001	000000	000000	000000	000000
000000000002	000000	000000	000000	000000
00000000003	000000	000000	000000	000000
000000000004	000000	<u>0</u> 00000	000000	000000
00000000005	000000	000000	000000	000000
00000000000	000000	000000	000000	000000
00000000007	000000	000000	000000	000000
00000000010	000000	000000	000000	000000
00000000011	000000	000000	000000	000000
00000000012	000000	000000	000000	000000
00000000013	000000	000000	000000	000000
00000000014	000000	000000	000000	000000
00000000015	000000	000000	000000	000000
00000000016	000000	000000	000000	000000
00000000017	000000	000000	000000	000000
00000000020	000000	000000	000000	000000
00000000021	000000	000000	000000	000000
00000000022	000000	000000	000000	000000
00000000023	000000	000000	000000	000000
00000000024	000000	000000	000000	000000
00000000025	000000	000000	000000	000000
00000000026	000000	000000	000000	000000
00000000027	000000	000000	000000	000000
00000000030	000000	000000	000000	000000
00000000031	000000	000000	000000	0000000
00000000032	000000	000000	000000	0000000
00000000033	000000	000000	000000	0000000
00000000034	000000	000000	000000	0000000
00000000035	000000	000000	000000	0000000
00000000037	000000	000000	000000	0000000

2. Type the new value you want to place in the memory location. The entire word is highlighted, which enables you to change it. You can move through the highlighted word with the arrow keys.

In the following example, 000217 was typed at memory location 0000000005b:

Ø	2	Memo	ry – Ab	solute	
00	000000000	000000	000000	000000	000000
00	000000001	000000	000000	000000	000000
00	000000002	000000	000000	000000	000000
	000000003	000000	000000	000000	000000
	000000004	000000	000000	000000	000000
	000000005	000000	000217	000000	000000
	000000006	000000	000000	000000	000000
	000000007	000000	000000	000000	000000
	000000010	000000	000000	000000	000000
	000000011	000000	000000	000000	000000
	000000012	000000	000000	000000	000000
	000000013	000000	000000	000000	000000
	000000014	000000	000000	000000	000000
	000000015	000000	000000	000000	000000
	000000016	000000	000000	000000	000000
	000000017	000000	000000	000000	000000
	000000020	000000	000000	000000	000000
	000000021	000000	000000	000000	0000000
	000000022	000000	000000	000000	0000000
	000000023	000000	000000	000000	0000000
	000000025	000000	000000	000000	0000000
	000000025	000000	000000	000000	0000000
	000000027	000000	000000	000000	0000000
	000000030	000000	000000	000000	000000
	000000031	000000	000000	000000	000000
	000000032	000000	000000	000000	000000
	000000033	000000	000000	000000	000000
	000000034	000000	000000	000000	000000
00	000000035	000000	000000	000000	000000
00	000000036	000000	000000	000000	000000
00	000000037	000000	000000	000000	000000

3. Press and release the Return key, which updates memory. In this example, memory location 0000000005b changed from 000000 to 000217, as shown:

Q	Memo	ry – Ab	solute	
00000000000	000000	000000	000000	000000
0000000000000001	000000	000000	000000	000000
000000000002	000000	000000	000000	000000
000000000003	000000	000000	000000	000000
000000000004	000000	000000	<u>0</u> 00000	000000
000000000005	000000	000217	000000	000000
00000000006	000000	000000	000000	000000
00000000007	000000	000000	000000	000000
00000000010	000000	000000	000000	000000
00000000011	000000	000000	000000	000000
00000000012	000000	000000	000000	000000
00000000013	000000	000000	000000	
000000000014	0000000	0000000	000000	0000000
000000000015	000000	000000	000000	
000000000017	000000	000000	000000	0000000
000000000000000000000000000000000000000	000000	000000	000000	0000000
0000000000021	000000	000000	000000	0000000
000000000022	000000	000000	000000	0000000
000000000023	000000	000000	000000	000000
000000000024	000000	000000	000000	0000000
00000000025	000000	000000	000000	000000
00000000026	000000	000000	000000	000000
00000000027	000000	000000	000000	000000
00000000030	000000	000000	000000	000000
00000000031	000000	000000	000000	000000
00000000032	000000	000000	000000	000000
00000000033	000000	000000	000000	000000
00000000034	000000	000000	000000	000000
00000000035	000000	000000	000000	000000
00000000036	000000	000000	000000	000000
00000000037	000000	000000	000000	000000

4. Repeat Steps 1 through 3 to change additional memory locations.

Using the Keyboard Accelerator

The keyboard accelerator offers another way to change memory and includes several other features. To access the keyboard accelerator, move the cursor inside a memory window and press the spacebar. The MME Keyboard Accelerator window appears:



Dump Command

The first command in the MME Keyboard Accelerator window is the Dump command. The Dump command dumps data to a file or printer. When you type the letter D in the MME Keyboard Accelerator window, the window changes to:

Q	MME Keyboard Accelerator
Address:	[0-7]* - SPACE when complete
Dump to I	Printer Parcel

NOTE: By default, the Dump command dumps parcel data to the printer. To change this, press the backspace key several times to delete the default selections. Then, type the menu options you want.

Enter the starting address of the data block you want to dump and press the spacebar; for example, if you wanted to dump the data block starting at 100, you would enter **100**:

Q	MME Keyboard Accelerator
Length:	[0-7]* - RETURN when ready
Dump to	Printer Parcel 100

Then, enter the length of the data block you want to dump; for example, if you wanted to dump a block of 2000_8 parcels, you would enter **2000**:

Q	MME Keyboard Accelerator
Length:	[0-7]* - RETURN when ready
Dump to	Printer Parcel 100 2000

Finally, press the Return key to dump the data. The window displays the main menu again:

Q	MME Keyboard Accelerator								
Commands:	Dump	Enter	Go	Halt	Load	S ave	0-7	RETURN	

Enter Command

The second command in the MME Keyboard Accelerator window is the Enter command. The Enter command puts data into memory. When you type the letter E in the MME Keyboard Accelerator window, the window changes to:

Q				ard Accelerator	
Enter: Enter	e X change,	Auto,	address	[0-7]*[a-d]	

This window gives you three command options:

• Type the letter **X** in the Enter menu to enter exchange data; *this option is not implemented yet*:

Q	MME Keyboard Accelerator
Not implemented Enter eXchange	d – RETURN when ready

• Type the letter **A** in the Enter menu to start the automatic increment mode. Automatic increment mode enables you to enter data into consecutive memory locations without having to manually enter each memory address.

For example, if you wanted to enter data in consecutive memory locations starting at address 102, you would perform the following steps:

1. Enter the address at which you want to enter the data and press the Return key. For this example, you would enter **102**:

Q					Accelerator	
Data: Enter	RETURN when Auto 102	complete,	ESC t	0	cancel	

2. Enter the first item of data and press the Return key. For this example, you would enter **17777 177777**:

Q	MME Keyboard Accelerator				
Data:	RETURN when complete, ESC to cancel				
Enter	Auto 102 17777 1777				

The window advances to the next memory location:

Q	MME Keyboard Accelerator
Data:	RETURN when complete, ESC to cancel
Enter	Auto 103

3. Enter the next item of data and press the Return key. Repeat this process to enter all of your data. When you have finished entering data, press the Esc key to halt automatic increment mode. The MME Keyboard Accelerator window returns to the main menu:

 Ø
 MME Keyboard Accelerator

 Commands: Dump Enter Go Halt Load Save 0-7 RETURN

• Type a number-letter combination in the Enter menu to enter a parcel address that indicates the memory address you want to change. For example, if you wanted to change parcel 1000b, you would enter **1000b**:

Q	MME Keyboard Accelerator
Address: Enter 10	[O-7]*[a-d] - SPACE when complete

Then, enter the data you want to write to memory; for example, if you wanted to enter 177777 177777 177777 177777, you would enter **177777 177777 177777** 177777

Q	MME I	Keyboar	rd Accelerato	,
RETURN when Enter 1000b	177777	177777	177777	

Press the Return key to write the data to memory:

ſ	Q	······						
I	00000001000	000000 177777 177777 177777						
I	00000001001	77777 000000 000000 000000						
I	00000001002	000000 000000 000000 000000						

The window displays the main menu again:

Q	MME Keyboard Accelerator								
Commands:	Dump	Enter	Go	Halt	Load	Save	0-7	RETURN	

Go Command

The third command in the MME Keyboard Accelerator window is the Go command. The Go command runs the assigned control point(s). When you type the letter **G** in the MME Keyboard Accelerator window, the window changes to:

Q	MME Keyboard Accelerator
RETURN when ready Go Selected	

NOTE: By default, the Go command runs the selected control point(s). To run all control points, press the backspace key to move backward through the menu selections. Then, type the menu options you want.

Press the Return key to start the control point(s). The window displays the main menu again:

Q	MME Keyboard Accelerator								
Commands:	Dump	Enter	Go	Halt	Load	S ave	0-7	RETURN	

Halt Command

The fourth command in the MME Keyboard Accelerator window is the Halt command. The Halt command halts the currently running control point(s). When you type the letter H in the MME Keyboard Accelerator window, the window changes to:

Q	MME Keyboard Accelerator
RETURN	when ready
Halt Se	elected Xchange Dump

NOTE: By default, the Halt command halts the selected control point(s) and performs an exchange dump. To halt all control points, perform a register dump, or halt with no dump; press the Back Space key to move backward through the menu selections. Then, type the menu options you want.

Press the Return key to halt the control point(s). The window displays the main menu again:

Commands: Dump Enter Go Halt Load Save O-7 RETURN

Load Command

The fifth command in the MME Keyboard Accelerator window is the Load command. The Load command loads a control point or data set. When you type the letter L in the MME Keyboard Accelerator window, the window changes to:

Q	MME Keyboard Accelerator		
Load: Load	Control point, Data		

This window gives you two options: loading a control point and loading a data set.

• Type the letter **C** in the Load menu to load a control point.

Q	MME Keyboard Accelerator
File: RETU	RN when ready (rel/diag.cp01)
Load Contro	ol point ∎

Then, enter the name of the control point from the current offline diagnostic release that you want to load. For example, if you want to load asb.t, enter **asb.t**:

Q	MME Keyboard Accelerator
File:	RETURN when ready (rel/diag.cp01)
Load	Control point asb.t

Finally, press the Return key to load the control point. The window displays the main menu again:



• Type the letter **D** in the Load menu to load a data set; *this command is not implemented yet.*

Save Command

The sixth command in the MME Keyboard Accelerator window is the Save command. The Save command saves a control point or data set. When you type the letter \mathbf{S} in the MME Keyboard Accelerator window, the window changes to:

Q	MME Keyboard Accelerator
Save: Save	Control point, Data

This window gives you two options: saving a control point and saving a data set.

• Type the letter **C** in the Save menu to save a control point.

Q	MME Keyboard Accelerator
File:	RETURN when ready (usr)
Save	Control point

Then, enter a filename to save the control point. For example, to save the control point in the file named m_{yCP} , enter **mycp**:

Q	MME Keyboard Accelerator
File:	RETURN when ready (usr)
Save	Control point mycp

Finally, press the Return key to save the control point. The window displays the main menu again:

 Ø
 MME Keyboard Accelerator

 Commands: Dump Enter Go Halt Load Save 0-7 RETURN

• Type the letter **D** in the Save menu to save a data set; *this command is not implemented yet.*

Numeric Commands

Two options are available when you type a number in the MME Keyboard Accelerator window: you can display memory starting at the number, or you can convert the number from octal to parcel format or from parcel format to octal.

• To display a specific memory location, type the location in the MME Keyboard Accelerator window and press the Return key. For example, to view memory location 5000, enter **5000**:



The window displays memory at location 5000:

0	Memo	ry – Ab	solute	
00000005000	000000	000000	000000	000000
00000005001	000000	000000	000000	000000
00000005002	000000	000000	000000	000000
00000005003	000000	000000	000000	000000
00000005004	000000	000000	000000	000000
00000005005	000000	000000	000000	000000
00000005006	000000	000000	000000	000000
00000005007	000000	000000	000000	000000
00000005010	000000	000000	000000	000000

The window displays the main menu again:



• To convert a number from octal to parcel format or from parcel format to octal, type the number and press the spacebar. For example, to convert octal value 5000 to parcel format, enter **5000** and press the spacebar:



The window then displays the parcel format equivalent.

Q	MME Keyboard Accelerator
RETURN when done 5000 = 01200a	

Press the Return key, and the window displays the main menu again:

Q	MME Keyboard Accelerator								
Commands:	Dump	Enter	Go	Halt	Load	S ave	0-7	RETURN	

Return

To close the MME Keyboard Accelerator window, press the Return key while the cursor is in the window.

View -> CPUs -> Control Point



The View -> CPUs -> Control Point command, as shown at the left, causes the status information in the CPU selection, control point, and status area to display the control point name. Refer to "CPU Selection, Control Point, and Status Area" on page 51 for more information about the status information.

View -> CPUs -> Pass Count (Global)



The View -> CPUs -> Pass Count (Global) command, as shown at the left, causes the status information in the CPU selection, control point, and status area to display the total pass count for all sections. Refer to "CPU Selection, Control Point, and Status Area" on page 51 for more information about the status information.

View -> CPUs -> Pass Count (Section)



The View -> CPUs -> Pass Count (Section) command, as shown at the left, causes the status information in the CPU selection, control point, and status area to display the pass count for the current section. Refer to "CPU Selection, Control Point, and Status Area" on page 51 for more information about the status information.

View -> CPUs -> P Register



The View -> CPUs -> P Register command, as shown at the left, causes the status information in the CPU selection, control point, and status area to display the P register. Refer to "CPU Selection, Control Point, and Status Area" on page 51 for more information about the status information.

View -> CPUs -> CIP



The View -> CPUs -> CIP command, as shown at the left, causes the status information in the CPU selection, control point, and status area to display the current instruction parcel (CIP). Refer to "CPU Selection, Control Point, and Status Area" on page 51 for more information about the status information.

View -> Register Dump



The View -> Register Dump command, as shown at the left, enables you to view the contents of the registers that were dumped with the Register Dump option of the Halt button. This command displays the MME View Register Setup window:

🖉 🛛 MME View Register Setup								
Register:								
Exchange (W	/EXP)		V2					
Exchange (C	EXP)	VЗ						
Shared			V4					
B Registe	rs		V5					
T Registe		V6						
VO	VO			V7				
V1								
Format:								
Byte	Halfv	vord	Hex					
Parcel	Wor	d	Address					
CPU: ♥ 00 Cluster: ♥ 00 Size: ♥ Large Font: ♥ Medium View								

Perform the following procedure to manipulate this window:

1. Specify the register for which you want to view the register dump data. Click on Register: [Exchange (WEAR)], [Exchange (WEAR)], [Shared],

В	Registers	,	T Aegisters	,	VÜ	,	VI],		02	,
	Và	,	04],	V 5	,	VÉ		or	v7	_

- 2. Specify the format in which you want to view the dump data. Click on Format: [byte], [Halfword], [Hex.], [Pacel], [Word], Or [Address].
- 3. If you clicked on a CPU Register setting in Step 1, choose a value from the CPU: ☐ to specify the CPU for which MME should display register dump data.
- If you clicked on the setting in Step 1, choose a value from the Cluster:
 If you clicked on the stand
 If you clicked on the setting in Step 1, choose a value from the Cluster: If to specify the cluster of shared registers for which MME should display register dump data.

5. Choose the size of the display window from the Size: \Box .

The following window sizes are available:

Size	Description
Small	The window displays 10_8 words.
Medium	The window displays 20_8 words.
Large	The window displays 40_8 words.
X–Large	The window displays 1008 words.

6. Choose the font size you want to display in the window from the Font: ☑.

The following font sizes are available:

Size	Description
Small	The font size is small.
Medium	The font size is medium.
Large	The font size is large.
X–Large	The font size is extra large.

7. Click on <u>View</u>. MME displays a window that contains the specified register data.

View -> Error Log

(View ⊽)	
Memory	
CPUs	۵
Register Dump	
Error Log	
Memory Map	
Runtime Information	Δ
Listing	⊳
Notes	

The View -> Error Log command, as shown at the left, displays memory, register parity, shared register, logical address translation (LAT) table, and unknown errors logged in the status registers. Use this command to track the occurrence of these errors. This command displays the MME Error Log window:

Q		N	IME Error Lo	g		
View:	Memory Register	Shared LAT	Unknown	MEM S	HR RPE	
Clear) (Print ⊽) File: <u>u</u>	1.37	UKN			
NUM TYP	Memo E DESTINATION	ry Errors - E Bi	ntries O, En K SS SC MOL	rrors O D SYNDROME	LOG/PHY COUNT	Ŕ
		No Erro	rs Logged.			

An error indicator (MEM, SHR, RPE, LAT, or UKN) appears in bold type in this window when the corresponding type of error is logged. When you see an error indicator in bold type, click on the appropriate View setting.

The Clear button clears all logged errors from the display. The Print ∇ button sends a copy of the error log to the printer (choose Print -> To Printer) or to the file specified in the File field (choose Print -> To File).

Memory Error Information

Click on View: Number to view the memory error information. The MME Error Log window displays the following memory error information:

Label	Description
NUM	Entry number in the table
TYPE	Type of error that occurred (correctable or uncorrectable)

Label	Description
DESTINATION	Destination of the memory read error (cache read, V register read, S register read, A register read, T register read, B register read, fetch read, I/O read, exchange read, I/O write, processor write, reconfigure, or memory error)
SECT	Section where the error occurred
BANK	Bank where the error occurred
MODULE	Module where the error occurred
SYNDROME	Syndrome
LOG/PHY	Logical/physical CPU number
COUNT	Number of errors for this entry

Register Parity Error Information

Click on View: Argister to view the register parity error information. The MME Error Log window displays the following register parity error information:

Label	Description
NUM	Entry number in the table
TYPE	Type of error that occurred
CHIP	Chip number where the error occurred
LOG/PHY	Logical/physical CPU number
COUNT	Number of errors for this entry

Shared Register Error Information

Click on View: **Shared** to view the shared register error information. The MME Memory/Register Parity Error Log window displays the following shared register error information:

Label	Description
NUM	Entry number in the table
TYPE	Type of error that occurred
CLUSTER	Cluster where the error occurred

Label	Description

LOG/PHYLogical/physical CPU numberCOUNTNumber of errors for this entry

LAT Error Information

Click on View: LAT to view the LAT error information. The MME Error Log window displays the following LAT error information:

Label	Description
NUM	Entry number in the table
TYPE	Type of error that occurred
LOG/PHY	Logical/physical CPU number
COUNT	Number of errors for this entry

Unknown Error Information

Click on View: Unknown to view the unknown error information. (Unknown errors are errors that do not have the necessary bits set to identify the errors.) The MME Error Log window displays the following error information:

Label	Description
NUM	Entry number in the table
DATA	4 parcels of error data
COUNT	Number of errors for this entry

View -> Memory Map

Nemory	
CPus	ь
Register Dump	
Error toge	
Constant and the	
Avotime Information	ь
Listing	ь
NUCES	

The View -> Memory Map command, as shown at the left, displays a map of the contents of mainframe and SSD memory. This command displays the MME Memory Map window:

۲ <mark>۵</mark>		MME Memory Map	5
View:	Top Down	Bottom Up Available Allocated	
			Ţ
			-
			T
L			

The top scroll box in this window displays the contents of mainframe memory. The bottom scroll box in this window displays the contents of SSD memory.

You can manipulate the memory map by performing the following actions:

- Click on TopDown to view memory from the highest memory location to the lowest memory location.
- Click on to view memory from the lowest memory location to the highest memory location.
- Click on **Available** to view memory that is not currently used.
- Click on *Allocand* to view memory that is currently used.

Figure 7 shows an example memory map that is configured to display both allocated and available memory from the lowest available memory location to the highest available memory location.

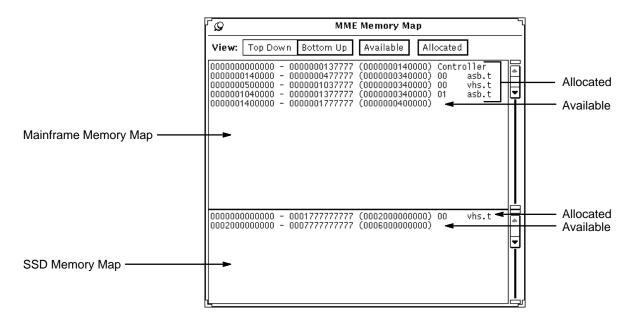


Figure 7. Memory Map Example

View -> Runtime Information -> Current



The View -> Runtime Information -> Current command, as shown at the left, displays the runtime information display for the current control point:

Ø						Run	time	In	Form	atio	n Die	snla	ıy - (10 v h	ic t		
MAIN	_	RROR SP CH		AGIN		PARA	METE	RS	CONT				· .	ANGE	Pas Err	s or tion	
		ch/Ex tor P							0000 0000						Con		000000000
	Cha 	nnels	Ru	nnin	g : 												
	20 1	21 1	22 1	23 1	24 1	25 1	26 1	27 1	30 1	31 1	32 1	33 1	34 1	35 1	36 1	37 1	
MA	IN -	Run	time	e dis	play												

Use this window to get updated information from the control point as the control point executes. Table 3 describes the runtime information categories you can view for control points.

On color displays, this window uses the following colors: black indicates that the text is mainframe memory data, blue indicates that the text is window text, and green indicates that you can click on the text to switch to a different display. On monochrome displays, bold text indicates that you can click on the text to switch to a different display.

Category	Description
MAIN	This display shows general information about the control point. This display typically includes the pass count, error count, current section, current condition, current subcondition, and control-point-specific information.
ERROR	This display shows any control-point-specific error information.
DIAGINFO	This display shows information from the standard locations, including the difference of the actual and expected values, the actual value, the expected value, the error count, the pass count, the error return address, the failing data address, the failing element mask, the section being tested, the condition being tested, the subcondition being tested, the condition loop count, and the subcondition loop count.
PARAMETERS	This display shows any control-point-specific parameters.
CONTENTS	This display shows the location of several standard locations, the code block, the data area, and the block storage segment.
HELP	This display shows any available help information.
EXCHANGE	This display shows information from the deadstart exchange package, starting exchange package, interrupt exchange package, working exchange packages, current exchange packages, and trap exchange packages.

Table 3. Control Point Runtime Information Categories

View -> Runtime Information -> Controller (Environment 2 Only)



The View -> Runtime Information -> Controller command, as shown at the left, displays the runtime information display for the controller:

Runtime Information Display - Controller N ERROR DIAGINFO PARAMETERS CONTENTS HELP EXCHANGE -037 040-077 100-137 140-177 CLUSTERS LIMITS
L > Pass Error SUT CUT P-reg IF EF Base 00 000000000 00000000 000 000 000 000000

Use this window to get updated information from the controller while the control points execute in environment 2. Table 4 describes the runtime information categories you can view for the controller.

On color displays this window uses the following colors: black indicates that the text is mainframe memory data, blue indicates that the text is window text, and green indicates that you can click on the text to switch to a different display. On monochrome displays, bold text indicates that you can click on the text to switch to a different display.

Category	Description
MAIN	This display shows general CPU information. This display contains the information used to update the MME base window (pass count, error count, section being tested, condition being tested, P register value, interrupt flags, and base address).
ERROR	This display shows the hartBEAT and idleSTAT parameters and interrupt flags for all CPUs. For more information about these parameters, refer to the online listing of the controller or to the <i>MME User Guide</i> , publication number HDM-102-A.
DIAGINFO	This display shows the pass count, error count, P register value, interrupt flags, mmeCIFM parameter, and dcCIFM parameter for the online listing of the controller or to the <i>MME User Guide</i> , publication number HDM-102-A.
000 – 037	This display shows the reservation table for channels $000 - 037$.
040 – 077	This display shows the reservation table for channels $040 - 077$.
100 – 137	This display shows the reservation table for channels 100 – 137.
140 – 147	This display shows the reservation table for channels 140 – 147.
CLUSTERS	This display shows the cluster reservation table.
LIMITS	This display shows the control point base and limit values for each CPU that is running control point code.
PARAMETERS	This display shows the MWS-to-CPU and CPU-to-MWS request and response parameters.
CONTENTS	This display shows the location of several standard locations, the code block, and the block storage segment.
EXCHANGE	This display shows information from the controller's deadstart exchange package, starting exchange package, interrupt exchange package, working exchange packages, trap exchange packages, and buffer exchange packages. This display also shows the trap exchange address (XA) and trapSTAT parameter. For more information about these parameters, refer to the online listing of the controller or to the <i>MME User Guide</i> , publication number HDM-102-A.

Table 4. Controller Runtime Information Categories

View -> Listing -> Current



The View -> Listing -> Current command, as shown at the left, displays the listing for the current section in the Sections scroll box. Use this command to learn more about the current section; the listing describes what is stored in memory and what the section is testing. This command displays the listing in a separate window that resembles the following window.

	rel/diag/cach.t/lst/cach00.t.l.Z
(Find Forward) (Find Backward) Patte	ern: (STDLOC) (PARAM) (CODE V) (IDATA) (UDATA)
1CAL Version 2 - 9.0ed 04 (06/23/94)	<pre>mmefrm400 10. 11. 12. 12. 13.* 14.* Cray Research, Inc. 15.* Unpublished Proprietary Information - All Rights Reserved. 16.* 17.**********************************</pre>

You can manipulate the listing as follows:

- To search forward from the current location in the window for a data pattern, enter the pattern in the Pattern field and click on (Find Forward).
- To search backward from the current location in the window for a data pattern, enter the pattern in the Pattern field and click on
 (Find Backward).
- To split the window to view multiple areas of the listing, press the MENU mouse button over the scroll bar and choose **Split View**. Choose **Join Views** to restore the window to the original view.
- To make the window a full-size display, double click in the window header. This enables you to see more of the listing at one time. Double click in the window header to return the window to normal size.
- To view the standard locations, parameters, initialized data (IDATA), or uninitialized data (UDATA); click on the corresponding button ((STDLOC), (PARAM), (IDATA), or (UDATA)).
- To view the standard code, choose CODE -> STDCODE; to view the main code, choose CODE-> MAIN; or to view the subroutines, choose CODE -> CODESUB.

View -> Listing -> Controller



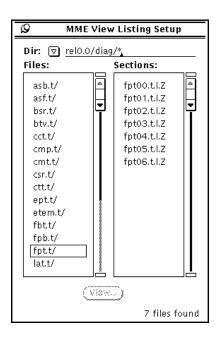
The View -> Listing -> Controller command, as shown at the left, displays the listing for the diagnostic controller. Use this command to learn more about the diagnostic controller. This command displays the listing in a separate window.

Refer again to the previous subsection, "View Listing -> Current," for an example of the listing window and a description of how to manipulate the listing.

View -> Listing -> Other



The View -> Listing -> Other command, as shown at the left, displays the listing for the control point section you specify. Use this command to get detailed information about a control point section. This command displays the MME View Listing Setup window:



Each section is an individual file, so you must specify the control point and then pick one section. To do this, perform the following steps. 1. In the Dir field, specify the directory in which to search for the listing file; enter the directory in the Dir field or choose it from the Dir: 🔽.

The Dir: v includes selections for the usr diagnostic directory, all of the rel directories, and all of the alpha directories. The RESET CHOICE DEFAULTS selection resets the field to the directory you selected with the Arkare, Alpha, User, Diag, Unility, Chay, and IEEE settings.

- 2. Click on the control point in the Files scroll box. The Sections scroll box displays the section listings available.
- 3. Click on the section in the Sections scroll box for which you want to see the listing.
- 4. Click on <u>View...</u>). MME displays the listing.

Refer again to the "View Listing –> Current" subsection on page 107 for an example of the listing window and a description of how to manipulate the listing.

View -> Notes

View 🗸	
Memory	
CPUs	⊳
Register Dump	
Error Log	
Memory Map	
Runtime Information	⊳
Listing	⊳
Notes	D

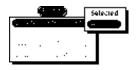
The View -> Notes command, as shown at the left, displays the MME release notes in a separate window. Use this window to read about any changes to MME for the current offline diagnostic release.

Edit -> Delete Control Point -> Selected



The Edit -> Delete Control Point -> Selected command, as shown at the left, deletes the current control point selected in the Control Points scroll box. Use this command to delete a control point you are no longer using.

Edit -> Delete Control Point -> All



The Edit -> Delete Control Point -> All command, as shown at the left, deletes all loaded control points. Use this command when you are done using all control points.

Edit -> Assign CPU(s)

This feature has not been implemented yet.

Edit -> Deassign CPU(s)

This feature has not been implemented yet.

Properties -> Environment -> ENV2 (Environment 1 Only)



The Properties -> Environment -> ENV2 command, as shown at the left, switches MME to environment 2. Use this command to switch to multiple-control-point testing. This command is available in environment 1 only.

Properties -> Environment -> ENV1 (Environment 2 Only)



The Properties -> Environment -> ENV1 command, as shown at the left, switches MME to environment 1. Use this command to switch to single-control-point testing. This command is available in environment 2 only.

Properties -> Environment -> ENV0



The Properties -> Environment -> ENV0 command, as shown at the left, switches MME to environment 0. Use this command to switch to MWS-based testing. Refer to the "Environment 0" section earlier in this document for more information about environment 0.

Properties -> Partition



The Properties -> Partition command, as shown at the left, selects the logical partition in which MME will run. Use this command to select the partition you want to troubleshoot. MME scans the current configuration for available partitions and allows you to select only partitions that allow maintenance or concurrent maintenance. MME displays the available partitions in a menu attached to the Properties -> Partition menu command.

Properties -> Resource Allocation



The Properties -> Resource Allocation command, as shown at the left, changes the way MME performs. Use this command to change the memory allocation, the CPU automatic assignment option and CPU modes; the CPU-to-memory delays; or the section swap interval. This command displays the MME Resource Allocation window; choose the category that you want to modify from the Category: \bigtriangledown .

Changing Mainframe Memory Allocation and SSD Memory Allocation

To change the mainframe memory allocation (in environment 2) or SSD memory allocation (in environment 1 or 2), choose **Mainframe/SSD Memory Allocation** from the Category: . The MME Resource Allocation window changes to:

Q	MME Resource Allocation				
Catagory: [▼ Mainframe/SSD Memory Allocation				
MAINFRA	MAINFRAME MEMORY ALLOCATION				
Allocatio	n Mode:				
Bottom U	Jp Top Down Random Partition				
P: SSD MEM	Partition Count: <u>4</u> (A) Partition Size: <u>0000000340000</u> SSD MEMORY ALLOCATION Address Parameters				
	000000000000				
u v	000100000000				
Count:	Partition Parameters <u>4</u> 0000200000000				

Mainframe Memory Allocation

Mainframe memory allocation defines how control point sections are loaded into mainframe memory (refer to Figure 8), whether mainframe memory is partitioned, the number of memory partitions, and the size of memory partitions. Mainframe memory allocation changes affect only the control points that are loaded after the changes are made.

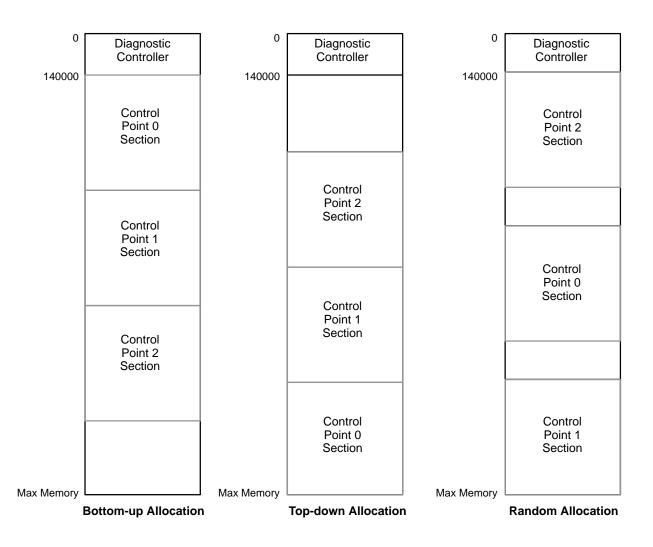


Figure 8. Mainframe Memory Allocation in Environment 2

You can specify the allocation mode by clicking on $\mathbb{E}_{\text{Ation } U\mu}$, $\mathbb{T}_{\text{OpDOwn}}$, or $\mathbb{R}_{\text{Ation}}$. You can also configure memory into partitions that cause all control points to use the same amount of memory; to do this, click on $\mathbb{P}_{\text{Ation}}$. When you partition memory, the number of control points you may load is limited to the number of partitions available.

Partitions are defined by count and size. The easiest way to define partitions is to enter the desired number of partitions in the Partition Count field. MME automatically calculates the size of the partitions. You may also define partitions by entering the desired partition size in the Partition Size field. MME automatically calculates the number of partitions. The default partition count is the number of CPUs MME uses.

MME uses the following formulas to calculate the partition size and partition count; the diagnostic_controller_size is 140000₈ words.

- Partition_size = (mainframe_memory_size minus diagnostic_controller_size) divided by partition_count
- Partition_count = (mainframe_memory_size minus diagnostic_controller_size) divided by partition_size
- **NOTE:** If you use these formulas to determine a partition count or size, ensure that you use all octal values.

SSD Memory Allocation

SSD memory allocation defines where in SSD memory control points can be loaded and how SSD memory should be partitioned. The SSD MEMORY ALLOCATION portion of this window contains the SSD memory allocation parameters. SSD memory allocation changes affect only the control points that you load after the changes are made.

The Address Parameters specify the block of SSD memory in which MME can load control points. The Start value specifies the beginning of the block in SSD memory. The Length value specifies the length of the block. The Limit value specifies the last available memory address for the block. The size of the SSD you have configured in SCE determines the Start, Length, and Limit values; you cannot modify these values in the MME Resource Allocation window.

The Partition Parameters, which are available in environment 2 only, configure SSD memory into partitions that cause all control points to use the same amount of SSD memory. When you partition SSD memory, the number of control points you may load in SSD memory is limited to the number of partitions available.

SSD memory partitions are defined by count and size. The easiest way to define partitions is to enter the desired number of partitions in the Count field. MME automatically calculates the size of the partitions. You may

also define partitions by entering the desired partition size in the Size field. MME automatically calculates the number of partitions. The default partition count equals the number of CPUs that MME is using.

MME uses the following formulas to calculate the partition size and partition count:

- Partition_size = SSD_memory_size divided by partition_count
- Partition_count = SSD_memory_size divided by partition_size

NOTE: If you use these formulas to determine a partition count or size, ensure that you use all octal values.

Changing the Auto Assignment, CPU Allocation, or CPU Control Options

To change the CPU assignment option and CPU modes, choose **CPU Allocation/Control** from the Category: ☑. The MME Resource Allocation window changes to:

Ø MMER	esou	'ce A	lloca	tion	
Catagory: 🔽 CPU Allocation/Control					
Auto Assignment:	Disa	bled	Enal	oled	
CPU Mode:					
I/O CPU		00	1.)	20	
Cache Disable		01	11	121	31
IRP Disable		02	12	<u> </u>	
IUM Disabled		03	13	123	
ICM Disabled		-)4	14	h	34
SBC/DBD Disabled		05	15		35
SEC/DED Disabled		-36	16	28	38
		hannand	ļ		
		07	17		
			(1744	9910)	
PCI: 00000004000					

Perform the following procedure to manipulate this window:

1. Specify the Auto Assignment option. Click on **Dividued** or **Enabled** to specify whether MME automatically assigns the first available CPU to a control point when the control point is loaded.

2. Click on the CPU Mode allocation option you want to change:

Mode	Description
1/0 CPu	Specifies a CPU to read and write memory
Cache Disable	Disables scalar cache
IAP Dicable	Disables interrupt on register parity error (IRP) for a CPU by clearing the IRP flag in the starting exchange package for the CPU if the flag is set
IUM Disabled	Disables interrupt on uncorrectable memory error (IUM) for a CPU by clearing the IUM flag in the starting exchange package for the CPU if the flag is set
ICM Disabled	Disables interrupt on correctable memory error (ICM) for a CPU by clearing the ICM flag in the starting exchange package for the CPU if the flag is set
SBC 4080 Likabled	Disables single-byte correction/ double-byte detection (SBCDBD)
SEC /DED Likabled	Disables single-error correction/ double-error detection (SECDED)

- Click on the CPU(s) that you want to set for the selected mode. Only one CPU can be selected as the I/O CPU. The other modes can be set for one or more CPU(s) at a time. Use the Toggle button to toggle the CPU selections, except in Troggle mode.
- 4. In environment 2 only, in the PCI field, enter the number you want to store in the PCITIME standard location. This value specifies how often the CPU checks the controller communication port to determine if MME has a function for the CPU to perform.

Changing the CPU-to-Memory Delay

To change the CPU-to-memory delays, choose **CPU/Memory Delay** from the Category:

The MME Resource Allocation window changes to:

& MME Resource Allocation									
Catago	Catagory: 🔽 CPU/Memory Delay								
Set D	Set Delay For: All CPUs Single CPU								
CPUs	:			Delay	/ (C	P):			
00	10	20	30	CJO:	0	4	16	63	
(11	11	21	31	CJ1:	0	4	16	63	
(.2)	12	22	32	CJ2:	0	4	16	63	
03	13	23	33	CJ3:	0	4	16	63	
(4	1.4	2-4	3-4	CJ4:	0	4	16	63	
05	15	25	35	CJ5:	0	4	16	63	
06	16	26	36	CJ6:	0	4	16	63	
07	17	27	37	CJ7:	0	4	16	63	
Set All 0 CP Set All 4 CP Set All 63 CP									

Perform the following procedure to manipulate this window:

- 1. Specify the CPUs for which you want to set the delay (click on [All CPUs] to set the delay for all CPUs or click on [simile CPU] and a CPU number to set the delay for one CPU).
- Specify the delay in clock periods (CPs) for each of the ports (CJ0 through CJ7; CJ refers to the option type), or click on (Set All 0 CP), (Set All 4 CP), (Set All 15 CP), or (Set All 63 CP) to set the delay for all ports to the same value.

Changing the Section Swap Interval and Enabling or Disabling the Maximum Pass Option

To change the section swap interval and enable or disable the maximum pass option, choose **Control Point Properties** from the Category: \bigtriangledown . The MME Resource Allocation window changes to:

Q	MME Resource Allocation
Catagory: 🔽) Control Point Properties
Swap Inter	val: <u>100</u> [] [] [] (msec)
Maximum I	Pass: Disabled Enabled

Section Swap Interval

The section swap interval value specifies how often MME checks the pass and error counts for each section to determine whether the section should be swapped. To change this interval, enter a new value in the field or move the slider to select a value.

Maximum Pass Option

The maximum pass option specifies whether MME should stop executing a control point section when the pass count for the section reaches the maximum pass value. Click on the appropriate setting to enable or disable the option.

Modifying the Spare-chip Table that MME Uses

To modify the spare-chip table that MME uses, choose **Spare Chips** from the Category: \bigtriangledown . The MME Resource Allocation window changes to:

Q	MME Resource Allocation			
Catagory: 🔽	7 Spare Chips			
Currently the sparechip table contains the only the configured spares.				
Mode:	Reset – Use Configured Spares			
	Merge With Configured Spares			
	Overide Configured Spares			
Upper Barl Bit Corte: 🗐 ((((- Bits 48 0 0)) Lower Barl Bit Corte: 🗐 ((((- Bits 16 0)) Apply Selections				

Use this window to modify the spare-chip table that MME uses. This enables you to test the mainframe with different configurations of spare memory chips.

To use this window, perform the following procedure:

1. Click on a Mode. The Mode settings perform the following functions when you click on (Apply Selections):

Setting	Description
Reset – Use Conlinued Suares	MME writes a spare-chip table to the mainframe that contains only the spare chips configured in SCE.
Norge with Conligued Spaces	MME writes a spare-chip table to the mainframe that contains only the bad bit codes that you specify with the Upper Bad Bit Code: ⊽ and Lower Bad Bit Code: ⊽ menus. (This setting overrides the spare-chip selections in SCE.)

Setting

Overide Configured Spares

Description

MME writes a spare-chip table to the mainframe that merges the spare-chip table configured in SCE with the bad bit codes that you specify with the Upper Bad Bit Code: ♥ and Lower Bad Bit Code: ♥ menus.

- 2. From the Upper Bad Bit Code: ♥ and Lower Bad Bit Code: ♥, choose the bad bit codes that you want to use. (These menus are available for the ware with conliqued Space and Overide Configured Spaces settings.)
- 3. Click on (Apply Selections); MME writes the spare-chip table to the mainframe.

Ensure that you reset the spare-chip table when you are done using a special spare-chip table configuration. You can reset the spare-chip table by clicking on Mode: <u>Meret-use Configured Sumer</u> and then clicking on <u>(Apply Selections</u>). You can also reset the spare-chip table by applying the configuration in SCE.

Changing the I/O Maintenance Settings (Environment 1 Only)

To change the I/O maintenance settings in environment 1, choose I/O **Maintenance** from the Category: \boxdot . The MME Resource Allocation window changes to:

Ø MME Resource Allocation
Catagory: 🔽 I/O Maintenance
I/O Channel: ⑦ All
Maintenance Mode:
121 Disable SEC/DED checking in CPU buffers
123 Disable SEC/DED errors from CPU buffers
130 Disable SEC/DED checking in I/O
131 Disable SEC/DED errors from I/O

The I/O maintenance settings enable you to set maintenance modes for specific channels.

Table 5 shows the I/O maintenance mode settings available for each type of channel in the mainframe. When you click on a setting, MME sends the corresponding loop controller function to the IO module that includes the channels selected from the I/O Channel: 🔽.

Table 5. I/O Maintenance Modes	5
--------------------------------	---

Code	Description		
	All Channels		
121	Disable SECDED in CPU:		
	This command specifies that data being read from the CPU's I/O buffers (coming from common memory) should not be checked for SECDED errors.		
123	Disable SECDED errors from CPU:		
	This command specifies that SECDED errors detected in the I/O buffers (of the CPUs) should not be reported on the error logger channel.		

Code	Description
	All Channels (continued)
130	Disable SECDED in I/O:
	This command specifies that data being returned to common memory should not be checked for SECDED errors.
131	Disable SECDED errors from I/O:
	This command specifies that SECDED errors detected for common memory read data should not be reported on the error logger channel.
	SLOSP Channels
121	Disable SECDED in CPU:
	This command specifies that data being read from the CPU's I/O buffers (coming from common memory) should not be checked for SECDED errors.
123	Disable SECDED errors from CPU:
	This command specifies that SECDED errors detected in the I/O buffers (of the CPUs) should not be reported on the error logger channel.
130	Disable SECDED in I/O:
	This command specifies that data being returned to common memory should not be checked for SECDED errors.
131	Disable SECDED errors from I/O:
	This command specifies that SECDED errors detected for common memory read data should not be reported on the error logger channel.
	LOSP Channels
116	Force parity bits:
	This command forces the LOSP output channel parity bits to 1's.
121	Disable SECDED in CPU:
	This command specifies that data being read from the CPU's I/O buffers (coming from common memory) should not be checked for SECDED errors.
123	Disable SECDED errors from CPU:
	This command specifies that SECDED errors detected in the I/O buffers (of the CPUs) should not be reported on the error logger channel.
130	Disable SECDED in I/O:
	This command specifies that data being returned to common memory should not be checked for SECDED errors.

Table 5. I/O Maintenance Modes (continued)

Code	Description
	LOSP Channels (continued)
131	Disable SECDED errors from I/O:
	This command specifies that data being returned to common memory should not be checked for SECDED errors.
	HISP Channels
120	Disable SECDED generation:
	This command specifies that the hardware should not generate checkbytes for data that is being written into common memory. This command has no effect on SECDED syndrome checking for data coming from the I/O channels.
121	Disable SECDED in CPU:
	This command specifies that data being read from the CPU's I/O buffers (coming from common memory) should not be checked for SECDED errors.
122	Disable SECDED error from generation:
	This command specifies that the hardware should not report any SECDED errors that it senses for I/O data coming from an I/O channel. [The errors could have occurred in the I/O channel or in the input data buffers (DR chips).]
123	Disable SECDED errors from CPU:
	This command specifies that SECDED errors detected in the I/O buffers (of the CPUs) should not be reported on the error logger channel.
127	Disable channel errors [WRITE]:
	This command specifies that the hardware should not sense or report channel errors that occur in the write path of the IO module.
130	Disable SECDED in I/O:
	This command specifies that data being returned to common memory should not be checked for SECDED errors.
131	Disable SECDED errors from I/O:
	This command specifies that data being returned to common memory should not be checked for SECDED errors.
137	Disable channel errors [READ]:
	This command specifies that the hardware should not sense or report channel errors that occur in the read path of the IO module.

Table 5. I/O Maintenance Modes (continued)

Code	Description		
	VHISP Channels		
120	Disable SECDED generation:		
	This command specifies that the hardware should not generate checkbytes for data that is being written into common memory. This command has no effect on SECDED syndrome checking for data coming from the I/O channels.		
121	Disable SECDED in CPU:		
	This command specifies that data being read from the CPU's I/O buffers (coming from common memory) should not be checked for SECDED errors.		
122	Disable SECDED error from generation:		
	This command specifies that the hardware should not report any SECDED errors that it senses for I/O data coming from an I/O channel. [The errors could have occurred in the I/O channel or in the input data buffers (DR chips).]		
123	Disable SECDED errors from CPU:		
	This command specifies that SECDED errors detected in the I/O buffers (of the CPUs) should not be reported on the error logger channel.		
130	Disable SECDED in I/O:		
	This command specifies that data being returned to common memory should not be checked for SECDED errors.		
131	Disable SECDED errors from I/O:		
	This command specifies that data being returned to common memory should not be checked for SECDED errors.		

Table 5. I/O Maintenance Modes (continued)

Viewing the Current Working Directory Settings and Changing the Error Logger Access, Concurrent Maintenance Check, and Debug Messages Settings

To view the current working directory settings and to change the error logger access, concurrent maintenance check, and debug messages settings, choose **Miscellaneous** from the Category: . The MME Resource Allocation window changes to:

S MME Reso	ource Allocation		
Catagory: 🔽 Miscellan	ieous		
Current Working Directory: /tmp_mnt/data/nova/cmedev/t32			
Error Logger Access:			
Nonexclusive Exclus	ive		
Concurrent Maintena	unce Check:		
Disabled Enabled			
Debug Messages:			
Control Points	Utilities		
Channel Functions	SCE & Reset		
Diagnostic Requests	Runtime Information		
Maintenance Modes	Diagnostic Actions		
Run System]		
L			

Current Working Directory

The Current Working Directory field shows the directory that MME uses to access all relative paths [paths that do not start with a slash (/)] shown in MME windows. MME appends all relative paths to this directory to determine the absolute path that MME uses to access files. For example, if a window displays rel/diag.cp02.4/* and this field displays /cri/cme/t32, MME uses the /cri/cme/t32/rel/diag.cp.02.4/* directory to access files. This field is read-only.

The information shown in this field is useful in the Systems Test and Checkout (STCO) environment in which technicians have access to remotely mounted directories of prereleased software. This information enables a technician to verify the paths used to load and save files so the technician can ensure that the proper software is being used.

Error Logger Access

The Error Logger Access setting specifies whether MME has exclusive access to the error logger channel or whether MME allows other software to access the error logger channel. By default, MME has exclusive access to the error logger channel: when a control point requests MME to log errors from the error logger channel, MME restricts access to the error channel so the error logging software does not also log these known errors. The Error Logger Access settings enable you to specify whether MME has exclusive access to the error channel.

Click on Error Logger Access: **E**_{sclusive} to restrict access to the error logger channel so MME has exclusive access to the channel. Click on Error Logger Access: **P**_{sclusive} to enable other software to access the error logger channel.

Concurrent Maintenance Check

The Concurrent Maintenance Check setting specifies whether MME checks the concurrent mode (CONC) bit in the control point code before loading a control point section when MME is in concurrent mode. (MME is in concurrent mode when MME is running in a physical partition that has more than one logical partition or when MME is running in a logical partition that has CPUs in 256-Kword mode.)

If you click on Concurrent Maintenance Check: [foulded], MME checks the CONC bit and does not load control point sections that have the CONC bit set to 1 if MME is in concurrent mode.

CAUTION

Do not disable the concurrent maintenance check feature under normal operating conditions. If you do, MME can load control points in concurrent mode that could crash the operating system.

If you click on Concurrent Maintenance Check: Litabled, MME loads the control point sections without checking the CONC bit.

Debug Messages

The Debug Messages settings specify the amount of output that MME returns to the standard output window. Table 6 describes these settings.

Table 6. Environments 1 and 2 Debug Messa	ges Settings
---	--------------

Setting	Description
Control Points	MME displays information about control points (for example, when they are loaded and where in memory they are loaded).
Channel Functions	MME displays information about any channel functions that MME performs.
Diagnostic Requests	MME displays information about any diagnostic requests that occur.
Maintenance Modes	MME displays information about any maintenance modes that are set or changed.
Run System	MME displays information about any run system operations that occur.
Utilities	MME displays information about any actions that the utilities perform.
SCE & Reset	MME displays information about any SCE functions or reset activities that occur.
Runtime Information	MME displays information about any runtime information-related operations that occur.
Diagnostic Actions	MME displays information about actions that the diagnostics perform.

Properties -> Run System (Environment 2 Only)



The Properties -> Run System command, as shown at the left, enables you to enable and disable the run system and set the run system parameters.

The run system is an environment 2 operation mode that swaps CPUs among any eligible control points. This creates an operating system type of environment in which the control points simulate jobs. For a control point to be eligible, the control point must satisfy the following conditions:

• The control point must rotate under the run system. This property is defined by the programmer and communicated to MME through the section information.

The following control point sections rotate under the run system:

chn.t	lsc.t	rit02.t	srt03.e	srt07.e
chn.e	lsc.e	srt00.t	srt04.t	srt10.t
diag.t	patt00.t	srt00.e	srt04.e	srt10.e
diag.e	patt00.e	srt01.t	srt05.t	vhc.t
etem.t	patt01.t	srt01.e	srt05.e	vhc.e
etem.e	patt01.e	srt02.t	srt06.t	
find.t	rit00.t	srt02.e	srt06.e	
find.e	rit01.t	srt03.t	srt07.t	

• The control point must have only one CPU assigned to it.

This command displays the MME Run System window:

Q	MME Run System
Run Systen	n:
Disabled	Enabled
Mode:	
Swap Pair	Swap Group
Interval: <u>5</u> (seconds)	0 •[] 60

Perform the following procedure to manipulate this window:

1. Click on Run System: **Enabled** or **Ekabled** to enable or disable the run system.

You can enable the run system at any time. MME checks the currently loaded control points for eligible control points. If MME finds two or more eligible control points that have not detected errors, MME begins to swap the CPUs.

If MME cannot find at least two eligible control points without errors, MME idles the run system. The run system remains enabled and waits until at least two eligible control points without errors are available.

If a loaded control point does not rotate under the run system or is assigned more than one CPU, MME does not affect it.

- Specify the mode you want to use. Click on Mode: Swap Pair to swap CPUs for a pair of control points at each swap interval. Click on Mode: Swap GPUs for all control points at each swap interval.
- 3. Specify the interval (in seconds) that you want MME to delay before it swaps the CPUs. Enter the value in the Interval field or move the slider. An interval of 0 causes MME to swap CPUs as fast as possible.

Utilities -> Clear Mainframe



The Utilities -> Clear Mainframe command, as shown at the left, runs a utility in all CPUs to clear interrupts, registers, memory, and shared resources.

This command performs the following actions:

- 1. This command saves the current environment.
- 2. This command loads and executes the clr.t (or clr.e) section of the clr.t (or clr.e) utility in all CPUs.
- 3. This command loads the saved environment.

Utilities -> Clear SSD



Pattern... Find... Copy/Move...

Configuration... Logic Monitor... Command Buffer... **NOTE:** Before you can use this command, you must use SCE to add an SSD solid-state storage device (SSD) to your system configuration.

The Utilities -> Clear SSD command, as shown at the left, runs a utility to clear the very high-speed (VHISP) channels for the logical partition in which MME is running. This utility also clears any SSDs that are connected to the channels.

This command performs the following actions:

- 1. This command saves the current environment.
- 2. This command loads and executes the clrssd.t (or clrssd.e) section of the clr.t (or clr.e) utility in all CPUs.
- 3. This command loads the saved environment.

Utilities -> Pattern



The Utilities -> Pattern command, as shown at the left, enables you to fill a selected portion of mainframe memory with a data pattern. This command displays the MME Memory Pattern Utility window:

Ø M	IME Memo	ory Patter	rn Utility	
Mode:				
MWS	Alle	PUs		
CRAY	Single	-CPU (<u>(0)</u> (09)	
Save & Res	tae Envire	nment Or	n Completion	
Pattern Sel	ect:			
Zero	Zeros Even Bits			
Ones	5	Address		
Odd Bits U		User	er Defined	
User Defín	ed Format	b		
Byte	Partei	Balfword	w.xd	
User Defin			(.(.)	
			Q000000000Q	
	solute	-	000000000000000000000000000000000000000	
Ctrl	pt Base		000000000000000000000000000000000000000	
	Sta			

The pattern utility runs in two modes: the utility uses maintenance channel functions from the MWS to write the data pattern in mainframe memory, or the utility executes program code in one or more mainframe CPUs to write the data pattern into mainframe memory. The following subsections describe how to manipulate the MME Memory Pattern Utility window to use each of these modes.

Using the MWS to Pattern Mainframe Memory

Perform the following procedure to manipulate this window to have the MWS pattern mainframe memory:

1. Click on to use maintenance channel functions from the MWS to write the pattern to mainframe memory.

CAUTION

In environment 2, if you use the MWS to write the data pattern and if the memory you select to write with the pattern includes part of the diagnostic controller area (0 to 140000 octal), the maintenance channel functions will write the pattern over the diagnostic controller. If this occurs, you must use the Reset -> Server command to reload the diagnostic controller before you can run any control points.

2. Specify the pattern you want to use. The Pattern Select settings specify the following patterns:

Setting	Pattern
Zeros	The utility uses a pattern of all 0's.
Ones	The utility uses a pattern of all 1's.
Odd Bits	The utility uses a pattern that contains parcels of odd bits (125252 octal).
EvenBits	The utility uses a pattern that contains parcels of even bits (052525 octal).
Address	The utility uses a pattern that contains the address of each memory location that is being patterned.
user Delined	The utility uses a user-specified pattern. Specify the format (click on User Defined Format: Pro, Pacel, Halfword, or word). In the User Defined Pattern field, enter the pattern you want to use.
The addresses you enter in Stan 2 are either absolute (if you	

NOTE: The addresses you enter in Step 3 are either absolute (if you click on <u>Alusolute</u>) or relative to the base address of the current control point (if you click on <u>Curlot Base</u>).

- 3. Specify the area in memory that you want to write with the pattern. In the Start field, enter the first address that you want to write and then press the Return key. In the Length field, enter the length of the data block that you want to write and then press the Return key. In the Limit field, enter the last address that you want to write and then press the Return key.
 - **NOTE:** You must press the Return key after you enter data in these fields, or MME will not use the data. Entering data in two of the fields causes MME to automatically update the third field.
- 4. Click on <u>start</u>; MME writes the data pattern in mainframe memory.

Using One or More Mainframe CPUs to Pattern Mainframe Memory

The pattern utility can also execute control point code in one or more mainframe CPUs to pattern mainframe memory. With this option, MME loads the patt.t or patt.e utility into mainframe memory and executes the program code for the appropriate section in one or more CPUs.

The program code for patt.t or patt.e resides in the lower 040000_8 words of mainframe memory, so MME must use MWS-based patterning to write any portion of this memory after patt.t or patt.e has patterned the remaining memory. This has the following effects on the pattern utility:

- If the entire memory area that you want to pattern is located in the lower 040000_8 words, the pattern utility uses maintenance channel functions from the MWS to write the pattern to mainframe memory.
- If the starting address is less than 040000_8 and the memory area that you want to pattern exceeds memory address 040000_8 , the patt.t or patt.e code writes the pattern to mainframe memory above address 040000_8 . The pattern utility then uses maintenance channel functions from the MWS to write the pattern to mainframe memory below address 040000_8 .
- If the block of memory that you want to write with the data pattern is less than 040000₈ words in length, the pattern utility uses maintenance channel functions from the MWS to write the pattern to mainframe memory.

Perform the following procedure to manipulate the MME Memory Pattern Utility window to use one or more mainframe CPUs to pattern mainframe memory:

1. Click on to have one or all CPUs write the pattern. Then, perform one of the following actions.

• Click on All CPUs to have all CPUs write the data pattern to memory.

If MME is in environment 1, MME performs the following actions when you click on <u>start</u> (in Step 4 on page 136):

- a. MME loads the patt.t (or patt.e) utility.
- b. MME assigns all CPUs to the control point.
- c. MME executes the patt01.t (or patt01.e) section in all CPUs to write the data pattern to mainframe memory.

If MME is in environment 2, MME performs the following actions when you click on <u>start</u> (in Step 4 on page 136):

- a. MME switches to environment 1.
- b. MME loads the patt.t (or patt.e) utility.
- c. MME assigns all CPUs to the control point.
- d. MME executes the patt01.t (or patt01.e) section in all CPUs to write the data pattern to mainframe memory.
- e. When the patt01.t section has completed execution in all CPUs, MME switches back to environment 2 and reloads the diagnostic controller so you can execute control points again.
- Click on <u>simile CPU</u> to have one CPU write the data pattern to memory. (Specify the CPU in the CPU field.).

If MME is in environment 1, MME performs the following actions when you click on <u>start</u> (in Step 4 on page 136):

- a. MME loads the patt.t (or patt.e) utility.
- b. MME assigns the specified CPU to the control point.
- c. MME executes the patt01.t (or patt01.e) section in the selected CPU to write the data pattern to mainframe memory.

If MME is in environment 2, MME performs the following actions when you click on <u>start</u> (in Step 4 on page 136):

- a. MME switches to environment 1.
- b. MME loads the patt.t (or patt.e) utility.
- c. MME assigns the specified CPU to the control point.
- d. MME executes the patt01.t (or patt01.e) section in the selected CPU to write the data pattern to mainframe memory.
- e. When the patt01.t section has completed execution, MME switches back to environment 2 and reloads the diagnostic controller so you can execute control points again.
- 2. Specify the pattern you want to use. The Pattern Select settings specify the following patterns:

Setting	Pattern
Zenos	The utility uses a pattern of all 0's.
Ones	The utility uses a pattern of all 1's.
Odd Bits	The utility uses a pattern that contains parcels of odd bits (125252 octal).
EvenBits	The utility uses a pattern that contains parcels of even bits (052525 octal).
Address	The utility uses a pattern that contains the address of each memory location that is being patterned.

Setting

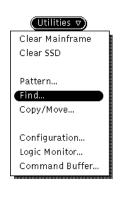
Pattern

User Delined

The utility uses a user-specified pattern. Specify the format (click on User Defined Format: [byw], [Pacel], [Halfwood], or [word]). In the User Defined Pattern field, enter the pattern you want to use.

- **NOTE:** The addresses you enter in Step 3 are either absolute (if you click on <u>Absolute</u>)) or relative to the base address of the current control point (if you click on <u>Curlet Base</u>)).
- 3. Specify the area in memory that you want to write with the pattern. In the Start field, enter the first address that you want to write and then press the Return key. In the Length field, enter the length of the data block that you want to write and then press the Return key. In the Limit field, enter the last address that you want to write and then press the Return key.
 - **NOTE:** You must press the Return key after you enter data in these fields, or MME will not use the data. Entering data in two of the fields causes MME to automatically update the third field.
- 4. Click on <u>start</u>; MME writes the data pattern in mainframe memory.

Utilities -> Find



The Utilities -> Find command, as shown at the left, searches mainframe memory for a data pattern. Use this command to locate all occurrences of a data pattern within a block of memory. This command displays the MME Find Utility window.

Q	MME Find Utility
Base:	
Absolute Relative	
Search Boundary:	
Byte Parcel Halfword Word	
Pattern/Mask Size:	
Word	
Pattern/Mask Format:	
Byte Parcel Halfword Word	
Pattern:	
000000 000000 000000 000000	
Mask: 177777 177777 177777 177777	
Address: 00000000000000000 Length: 00000000000000000	
Limit: 00000000000000	
(Start (Continue)	<u> </u>

The left side of the window contains the settings that specify the pattern to search for and the memory block to search. The right side contains a scroll box that displays memory locations with the matching data pattern. The scroll box displays up to 256 entries; if more than 256 matches are found, the message 256 matches, additional occurrences known to exist is displayed in the lower-left corner of the window. To view the additional occurrences, click on the <u>Continue</u> button.

Perform the following procedure to manipulate this window:

1. Specify the Base to use. For environment 1, Base should usually be set to Ausolule because only one control point is loaded.

Click on *Ausolute* to use memory addresses based at 0. Click on *Melative* to use memory addresses relative to the base address of the current control point section.

2. Specify the Search Boundary to use. The search boundary indicates the stride used for checking memory.

Click on **Prov** to check memory in byte increments, click on **Pacel** to check memory in parcel increments, click on **malfword** to check memory in halfword increments, or click on **ward** to check memory in word increments.

3. Specify the Pattern/Mask Size. The size indicates the size of the data pattern for which this utility searches and the mask it uses.

The settings available depend on the Search Boundary setting. Click on **Prop**, **Pacel**, **Halfwod**, or **word** to select the pattern and mask size.

4. Specify the Pattern/Mask Format. The format indicates the type of data pattern for which this utility searches and the mask it uses.

The settings available depend on the Pattern/Mask Size setting. Click on **Prim**, **Pacel**, **mattern**, or **word** to select the pattern and mask format.

- 5. Specify the data pattern you want to search for in the Pattern field.
- 6. Specify the mask you want to use in the Mask field. The mask specifies which bits to compare. If a bit in the mask is set to 0, the bit position is not compared; if a bit in the mask is set to 1, the bit position is compared.
- 7. Specify the memory block to search (performing any two of the following actions automatically updates the third field):
 - Enter the first address of the memory block in the Starting Address field and press the Return key.
 - Enter the length of the memory block in the Length field and press the Return key.
 - Enter the last address of the data block in the Limit field and press the Return key.
- 8. Click on <u>Start</u> to start the search. The <u>Start</u> button changes to <u>Stop</u>, and MME updates the MME Find Utility window.

ase:	00000000000100	000000 000000 000000 000000
Absolute Relative	00000000000101	000000 000000 000000 000000
Absolute helative	00000000000102	000000 000000 000000 000000
Search Boundary:	00000000000103	000000 000000 000000 000000
	00000000000104	000000 000000 000000 000000
Byte Parcel Halfword Word	00000000000105	000000 000000 000000 000000
	00000000000106	000000 000000 000000 000000
Pattern/Mask Size:	00000000000107	000000 000000 000000 000000
lulord	00000000000110	000000 000000 000000 000000
Word	00000000000111	000000 000000 000000 000000
Pattern/Mask Format:	00000000000112	000000 000000 000000 000000
Fatter II/ Mask Format.	00000000000113	000000 000000 000000 000000 000000 000000
Byte Parcel Halfword Word	000000000000114	
	000000000000115	
Pattern:	000000000000117	
000000 000000 000000 000000 .	000000000000120	
•••••		
Mask:	00000000000122	
177777 177777 177777 177777	00000000000123	
	- 00000000000124	
Address: 00000000000010.	00000000000125	000000 000000 000000 000000
	00000000000126	000000 000000 000000 000000
Length: <u>00000000100000,</u>	00000000000127	000000 000000 000000 000000
Limit: 00000000100010,	00000000000130	000000 000000 000000 000000
	00000000000131	000000 000000 000000 000000
(Stop) (Continue)	00000000000132	000000 000000 000000 000000
	00000000000133	000000 000000 000000 000000

Click on <u>Stop</u> to stop searching for the pattern; if more than 256 occurrences exist, click on <u>Continue</u> to see the next set of entries.

Utilities -> Copy/Move



The Utilities -> Copy/Move command, as shown at the left, enables you to copy (or move) data in mainframe memory. This command displays the MME Copy/Move Utility window:

G MME Co	py/Move Utility
~	
Mode:	
Parcel	Word
Base:	
Absolute C	Ctript Base
Source:	
Start:	000000000000
Length:	00000000000000000
Limit:	000000000000
Destination: Start:	00000000000000000
Сору	Move

Perform the following procedure to manipulate this window:

- 1. Specify the Mode. Click on reaction to use parcel values for addresses. Click on reaction to use word values for addresses.
- 2. Specify the Base of the addresses. Click on Ausolute to use absolute (based on 0) addresses. Click on Cirile Base to use addresses that are relative to the base address of the current control point.
- 3. Specify the location of the source data. In the Start field, enter the first address of the block of data that you want to move or copy. In the Length field, enter the length of the block of data to move or copy. In the Limit field, enter the last address of the block of data to move or copy.

You must press the Return key after you enter data in these fields, or MME will not use the data. Entering data in two of the fields causes MME to automatically update the third field.

4. In the Start field, specify the destination where MME should write the copied/moved data.

You must press the Return key after you enter data in this field.

5. Click on Copy to copy the data, or click on Move to move the data. (When you move data, MME fills the source mainframe memory locations with 0's.)

Utilities -> Configuration



The Utilities -> Configuration command, as shown at the left, starts the System Configuration Environment (SCE), which you use to configure the mainframe. For more information about SCE, refer to the *SCE User Guide*, publication number HDM-069-A.

Utilities -> Logic Monitor

Utilities Clear Mainframe Clear SSD Pattern... Find... Copy/Move... Configuration... Logic Monitor... Command Buffer...

The Utilities -> Logic Monitor command, as shown at the left, starts the Logic Monitor Environment (LME). For more information about LME, refer to the *LME User Guide*, publication number HDM-070-0.

Utilities -> Command Buffer

(Utilities ⊽)

```
Clear Mainframe
Clear SSD
Pattern...
Find...
Copy/Move...
```

Configuration... Logic Monitor... Command Buffer.. The Utilities -> Command Buffer command, as shown at the left, starts the Command Buffer Parser (CBP) application with the CRAY T90 series CBP runtime module. For more information about the CRAY T90 series CBP runtime module, refer to the *CBP Runtime Module* document, publication number HDM-071-A.

Reset -> Channel

Reset	\bigtriangledown
Channel	
Server	
Configura	ation
1/0 0	⊳
1/0 1	⊳
1/0 2	⊳
1/0 3	⊳
Shared O	
Shared 1	

The Reset -> Channel command, as shown at the left, resets the FY driver.

Reset -> Server

Reset	\bigtriangledown
Channel	
Server	
Configur	ation
1/0.0	⊳
I/O 1	
I/O 2	⊳
I/O 3	⊳
Shared O	
Shared 1	

The Reset -> Server command, as shown at the left, resets the server. This halts all control points, reloads the controller (environment 2 only), and reloads the control points.

NOTE: The control point reload function of the Reset -> Server command does not remove any global or local changes you have made to the control points. If you want to reload control points and remove any global or local changes, click on Reload.

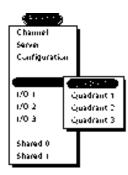
Reset -> Configuration

Reset	∇
Channel	
Server	
Configur	ation
I/O 0	⊳
I/O 1	⊳
1/0 2	⊳
1/0 3	⊳
Shared O	
Shared 1	

The Reset -> Configuration command, as shown at the left, causes SCE to reapply the configuration.

NOTE: This command does not work if any partitions have an OS owner. For more information about partition ownership, refer to the *SCE User Guide*, publication number HDM-069-A.

Reset -> I/O 0 -> Quadrant 0

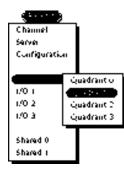


The Reset \rightarrow I/O 0 \rightarrow Quadrant 0 command, as shown at the left, uses logical IO module 0 to toggle Master Clear for I/O quadrant 0.

MME does not perform this command if either of the following conditions is true:

- The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)
 - **NOTE:** At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).
- More than one logical partition exists in the physical partition in which MME is running. (If MME toggled the Master Clear, all I/O activity for channels in use by the other logical partitions would also be stopped.)

Reset -> I/O 0 -> Quadrant 1



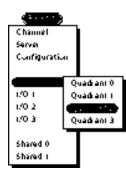
The Reset \rightarrow I/O 0 \rightarrow Quadrant 1 command, as shown at the left, uses logical IO module 0 to toggle Master Clear for I/O quadrant 1.

MME does not perform this command if either of the following conditions is true:

• The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)

NOTE: At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).

Reset -> I/O 0 -> Quadrant 2

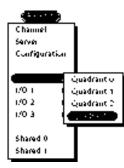


The Reset \rightarrow I/O 0 \rightarrow Quadrant 2 command, as shown at the left, uses logical IO module 0 to toggle Master Clear for I/O quadrant 2.

MME does not perform this command if either of the following conditions is true:

- The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)
 - **NOTE:** At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).
- More than one logical partition exists in the physical partition in which MME is running. (If MME toggled the Master Clear, all I/O activity for channels in use by the other logical partitions would also be stopped.)

Reset -> I/O 0 -> Quadrant 3



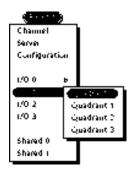
The Reset \rightarrow I/O 0 \rightarrow Quadrant 3 command, as shown at the left, uses logical IO module 0 to toggle Master Clear for I/O quadrant 3.

MME does not perform this command if either of the following conditions is true:

• The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)

NOTE: At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).

Reset -> I/O 1 -> Quadrant 0

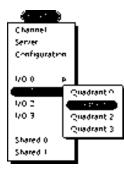


The Reset \rightarrow I/O 1 \rightarrow Quadrant 0 command, as shown at the left, uses logical IO module 1 to toggle Master Clear for I/O quadrant 0.

MME does not perform this command if either of the following conditions is true:

- The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)
 - **NOTE:** At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).
- More than one logical partition exists in the physical partition in which MME is running. (If MME toggled the Master Clear, all I/O activity for channels in use by the other logical partitions would also be stopped.)

Reset -> I/O 1 -> Quadrant 1



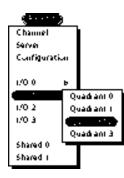
The Reset \rightarrow I/O 1 \rightarrow Quadrant 1 command, as shown at the left, uses logical IO module 1 to toggle Master Clear for I/O quadrant 1.

MME does not perform this command if either of the following conditions is true:

• The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)

NOTE: At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).

Reset -> I/O 1 -> Quadrant 2

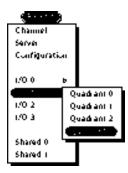


The Reset \rightarrow I/O 1 \rightarrow Quadrant 2 command, as shown at the left, uses logical IO module 1 to toggle Master Clear for I/O quadrant 2.

MME does not perform this command if either of the following conditions is true:

- The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)
 - **NOTE:** At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).
- More than one logical partition exists in the physical partition in which MME is running. (If MME toggled the Master Clear, all I/O activity for channels in use by the other logical partitions would also be stopped.)

Reset -> I/O 1 -> Quadrant 3



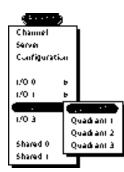
The Reset \rightarrow I/O 1 \rightarrow Quadrant 3 command, as shown at the left, uses logical IO module 1 to toggle Master Clear for I/O quadrant 3.

MME does not perform this command if either of the following conditions is true:

• The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)

NOTE: At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).

Reset -> I/O 2 -> Quadrant 0



The Reset \rightarrow I/O 2 \rightarrow Quadrant 0 command, as shown at the left, uses logical IO module 2 to toggle Master Clear for I/O quadrant 0.

MME does not perform this command if either of the following conditions is true:

- The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)
 - **NOTE:** At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).
- More than one logical partition exists in the physical partition in which MME is running. (If MME toggled the Master Clear, all I/O activity for channels in use by the other logical partitions would also be stopped.)

Reset -> I/O 2 -> Quadrant 1



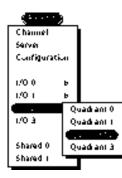
The Reset -> I/O 2 -> Quadrant 1 command, as shown at the left, uses logical IO module 2 to toggle Master Clear for I/O quadrant 1.

MME does not perform this command if either of the following conditions is true:

• The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)

NOTE: At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).

Reset -> I/O 2 -> Quadrant 2

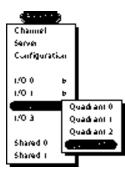


The Reset \rightarrow I/O 2 \rightarrow Quadrant 2 command, as shown at the left, uses logical IO module 2 to toggle Master Clear for I/O quadrant 2.

MME does not perform this command if either of the following conditions is true:

- The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)
 - **NOTE:** At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).
- More than one logical partition exists in the physical partition in which MME is running. (If MME toggled the Master Clear, all I/O activity for channels in use by the other logical partitions would also be stopped.)

Reset -> I/O 2 -> Quadrant 3



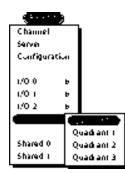
The Reset \rightarrow I/O 2 \rightarrow Quadrant 3 command, as shown at the left, uses logical IO module 2 to toggle Master Clear for I/O quadrant 3.

MME does not perform this command if either of the following conditions is true:

• The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)

NOTE: At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).

Reset -> I/O 3 -> Quadrant 0

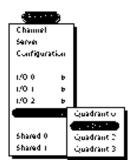


The Reset \rightarrow I/O 3 \rightarrow Quadrant 0 command, as shown at the left, uses logical IO module 3 to toggle Master Clear for I/O quadrant 0.

MME does not perform this command if either of the following conditions is true:

- The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)
 - **NOTE:** At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).
- More than one logical partition exists in the physical partition in which MME is running. (If MME toggled the Master Clear, all I/O activity for channels in use by the other logical partitions would also be stopped.)

Reset -> I/O 3 -> Quadrant 1



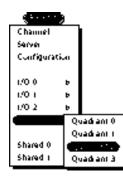
The Reset -> I/O 3 -> Quadrant 1 command, as shown at the left, uses logical IO module 3 to toggle Master Clear for I/O quadrant 1.

MME does not perform this command if either of the following conditions is true:

• The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)

NOTE: At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).

Reset -> I/O 3 -> Quadrant 2

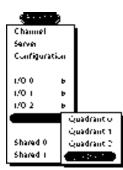


The Reset \rightarrow I/O 3 \rightarrow Quadrant 2 command, as shown at the left, uses logical IO module 3 to toggle Master Clear for I/O quadrant 2.

MME does not perform this command if either of the following conditions is true:

- The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)
 - **NOTE:** At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).
- More than one logical partition exists in the physical partition in which MME is running. (If MME toggled the Master Clear, all I/O activity for channels in use by the other logical partitions would also be stopped.)

Reset -> I/O 3 -> Quadrant 3



The Reset \rightarrow I/O 3 \rightarrow Quadrant 3 command, as shown at the left, uses logical IO module 3 to toggle Master Clear for I/O quadrant 3.

MME does not perform this command if either of the following conditions is true:

• The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)

NOTE: At the time of this printing, UNICOS does not support multiple logical partitions (creating logical machines).

Reset -> Shared 0

(Reset v	0
Channel	
Server	
Configurat	tion
1/0 0	⊳
1/0 1	⊳
1/0 2	⊳
1/0 3	⊳
Shared 0	
Shared 1	

The Reset -> Shared 0 command, as shown at the left, toggles Master Clear for shared module 0.

MME does not perform this command if either of the following conditions is true:

- The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)
- More than one logical partition exists in the physical partition in which MME is running. (If MME toggled the Master Clear, all I/O activity for channels in use by the other logical partitions would also be stopped.)

Reset -> Shared 1

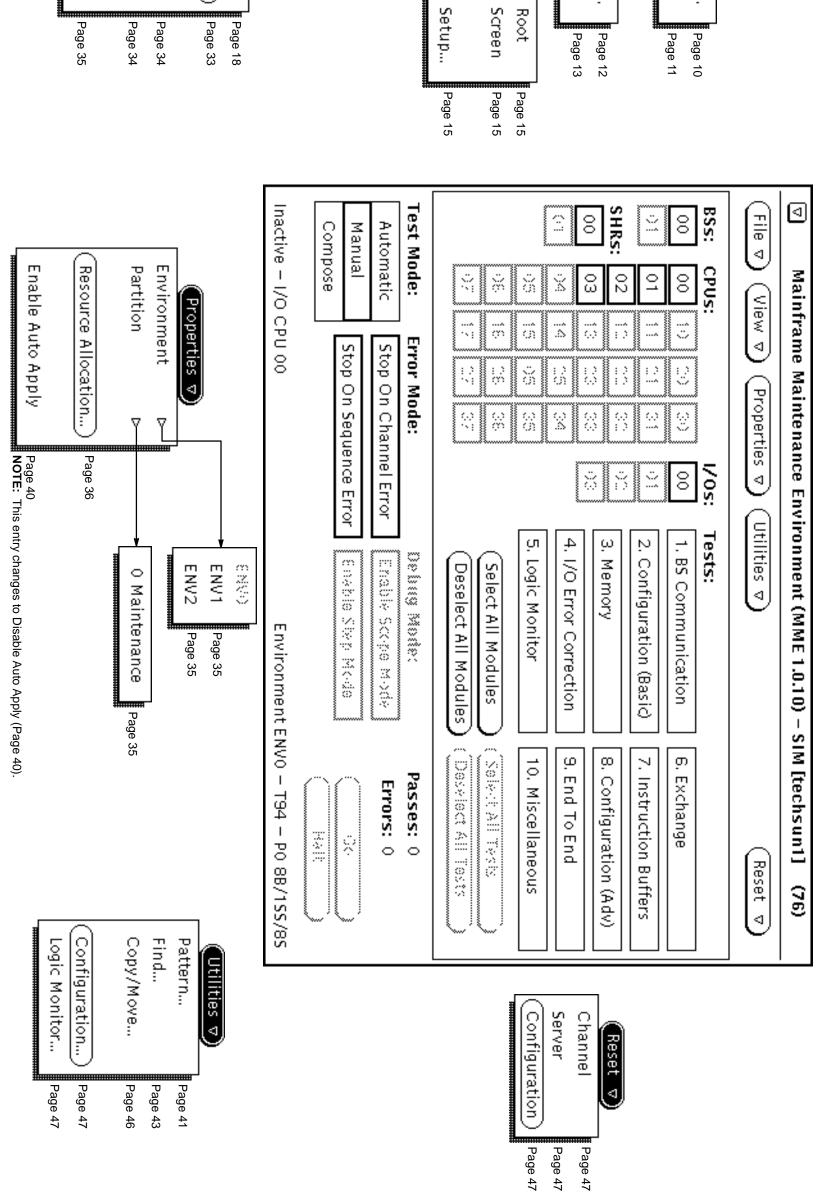
Reset	∇
Channel	
Server	
Configura	ation
1/0 0	⊳
I/O 1	
1/0 2	
1/0 3	⊳
Shared O	
Shared 1	

The Reset -> Shared 1 command, as shown at the left, toggles Master Clear for shared module 1.

MME does not perform this command if either of the following conditions is true:

- The operating system owns the logical partition in which MME is running, which implies concurrent maintenance is being performed. (If MME toggled the Master Clear, all I/O activity for channels in use by the OS would also be stopped.)
- More than one logical partition exists in the physical partition in which MME is running. (If MME toggled the Master Clear, all I/O activity for channels in use by the other logical partitions would also be stopped.)

Figure 9. Environment 0 Menu Quick Reference



Delete...

Page 14

Print 🖻

Dump...

Page 16

Save

V

Data...

Load

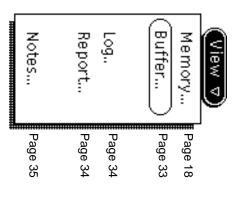
Y

Sequence...

File ⊽)

Data...

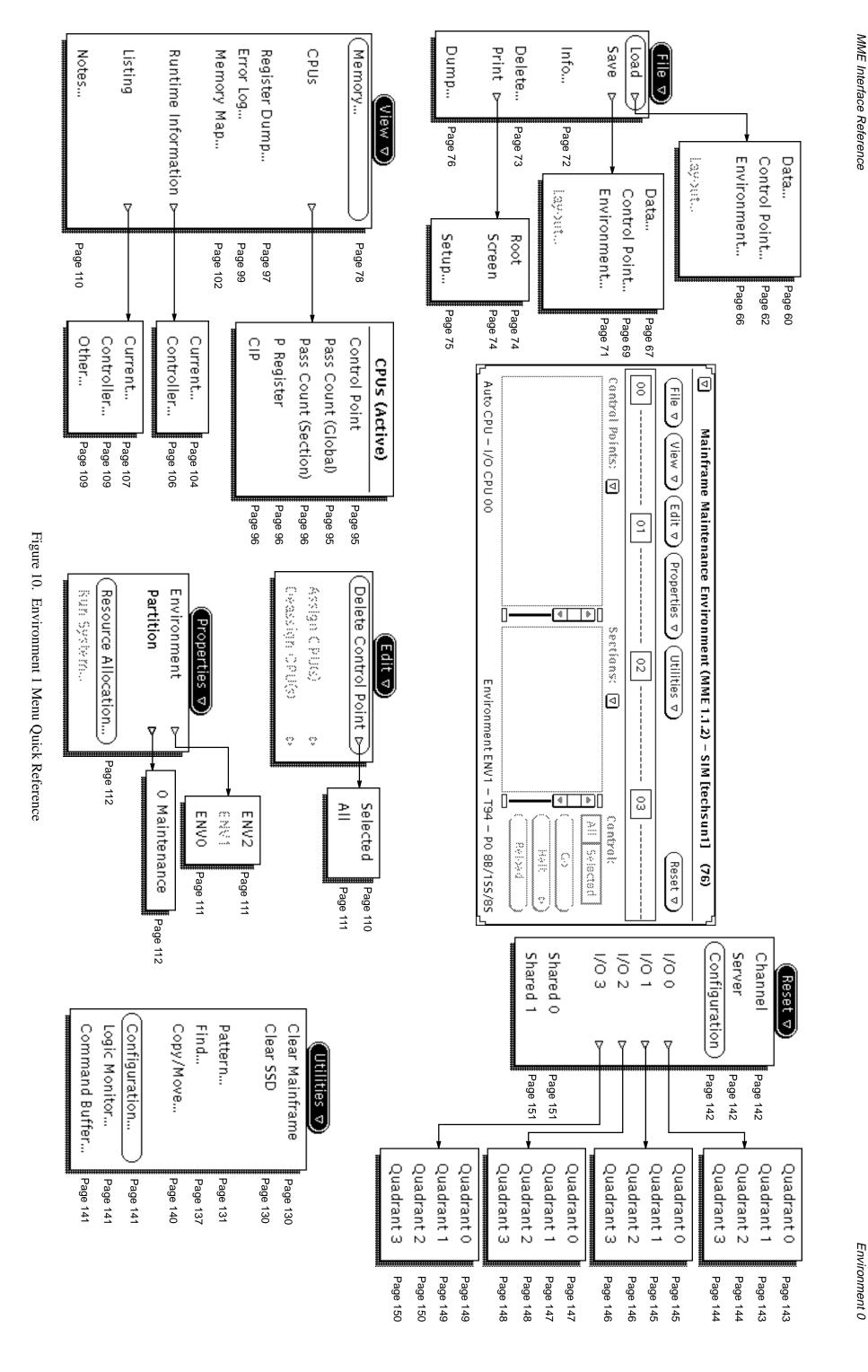
Sequence...





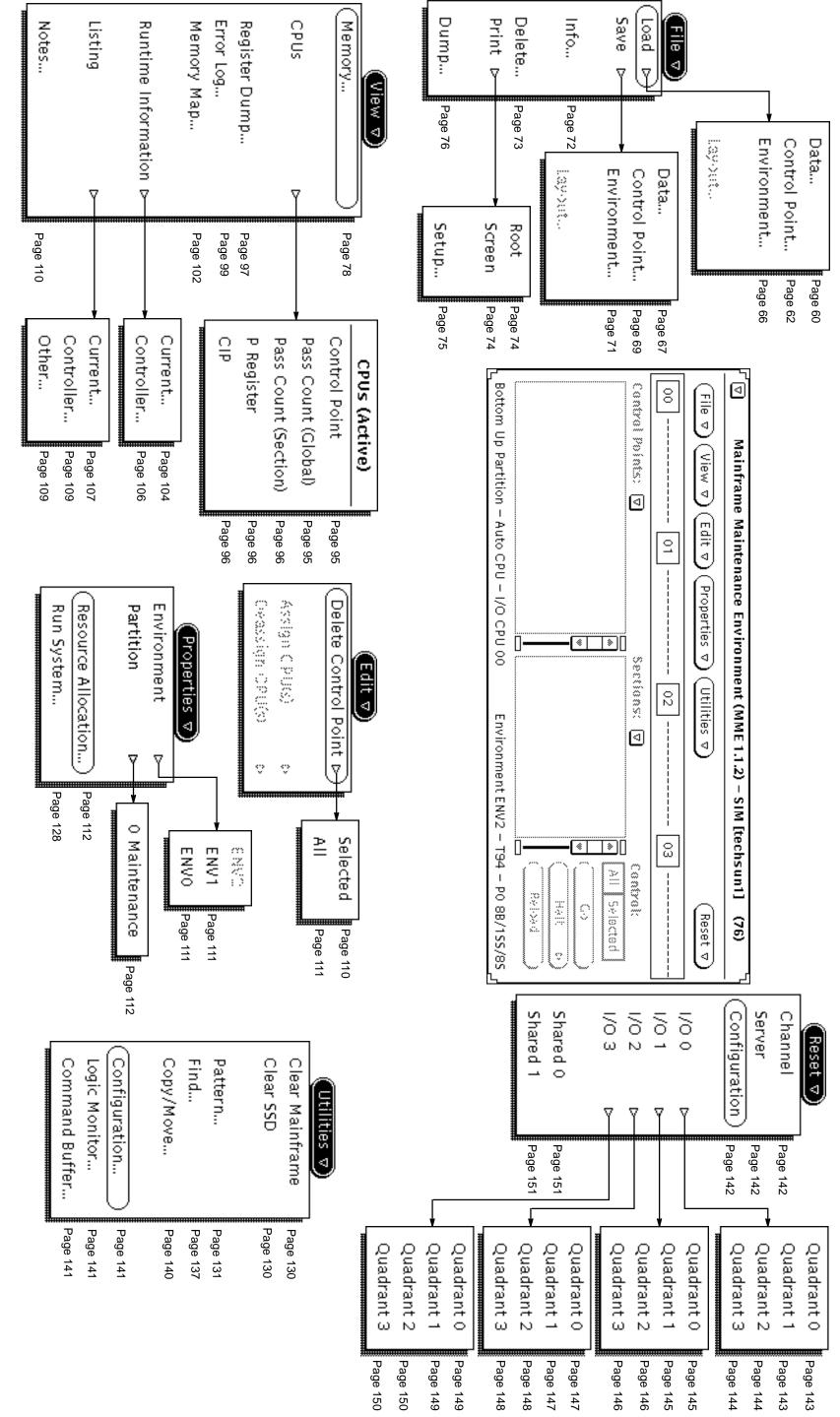
MME Interface Reference

Cray Research Proprietary



155

Cray Research Proprietary



HDM-008-A

Cray Research Proprietary

Figure 11. Environment 2 Menu Quick Reference

MME Interface Reference

÷,	
z	Utilities
÷.,	
2.	1.7
	1 👾
÷ι	1.22
511	10.
	10
	[⊲
3	\sim

157