

USD-III

Universal Symbolic Debugger for MICE-II and MICE-III Series

User's Guide

Doc. No. I49-000519
Second Edition
June 1991

MICROTEK INTERNATIONAL INC.

Development Systems Division

Asia Office:

6, Industry East Road 3
Science-based Industry Park
Hsinchu 30077, Taiwan, ROC
Tel: (35) 772155
Fax: (35) 772598

3300 N.W. 211th Terrace

Hillsboro, Oregon 97124-7136
U.S.A.
Tel: (503) 645-7333
Fax: (503) 629-8460

Europe Office:

Shiess-Strasse 72
D-4000 Düsseldorf 11
F.R. Germany
Tel: (211) 52-6070
Fax: (211) 59-6782

Trademarks Acknowledgement

IBM, PC, XT, AT, and PS/2 are trademarks of International Business Machines Corp.
MS-DOS and **MS-WINDOWS** are trademarks of Microsoft Corporation
MRI is a trademark of Microtec Research, Inc.
Intel is a trademark of Intel Corporation
NEC is a trademark of NEC Corporation

©1990 MICROTEK INTERNATIONAL INC. All rights reserved.

This manual is subject to change without notice. Nothing herein shall be construed as a recommendation to use any product in violation of existing patents or other rights of third parties.

THE USD-III AND THIS MANUAL ARE COVERED BY THE LIMITED WARRANTY SUPPLIED WITH THE SOFTWARE PACKAGE.

USD-III IS ALSO COVERED BY THE LICENSE AGREEMENT SUPPLIED FOR ALL MICROTEK PROPRIETY SOFTWARE PACKAGES. USD-III MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE TERMS OF THE LICENSE AGREEMENT.

Printed in Taiwan, ROC 6/91

CONTENTS

MICROTEK INTERNATIONAL INC.

1 INTRODUCTION

| | | |
|---------|--|------|
| 1.1 | WHAT IS USD-III ? | 1-1 |
| 1.2 | USD-III AT A GLANCE | 1-2 |
| 1.3 | INTRODUCTION TO MOUSE OPERATION | 1-4 |
| 1.4 | INTRODUCTION TO USD-III SCREEN LAYOUT | 1-5 |
| 1.4.1 | Menu Bar | 1-5 |
| 1.4.2 | USD-III Viewports | 1-6 |
| 1.4.3 | Pull-Down Menu | 1-7 |
| 1.4.4 | Pop-Up Dialog Box and Choice List | 1-9 |
| 1.4.5 | Message Line | 1-9 |
| 1.4.5.1 | Default Screen Message Line | 1-11 |
| 1.4.5.2 | Message Lines for Current Command | 1-13 |
| 1.4.5.3 | Message Line Status with Pull-Down Menu on Display | 1-14 |
| 1.4.5.4 | Message Line Status with Dialog Box on Display | 1-15 |
| 1.5 | ON-LINE <F1> "HELP" SUMMARY | 1-16 |

2 GETTING STARTED

| | | |
|-------|--------------------------------------|-----|
| 2.1 | WHAT YOU SHOULD GET | 2-1 |
| 2.2 | INSTALLATION AND BACKUP PROCEDURE | 2-1 |
| 2.3 | USD-III SETUP ON MS-DOS/PC-DOS HOSTS | 2-2 |
| 2.3.1 | RS-232C Serial Port Setup | 2-3 |
| 2.3.2 | Parallel Port Setup (MICE-III only) | 2-4 |
| 2.4 | INVOKING USD-III | 2-5 |
| 2.5 | QUITTING USD-III | 2-6 |
| 2.6 | SOFTWARE BREAKPOINT IMPLEMENTATION | 2-8 |

3 CONVENTIONS OF NOTATION AND COMMANDS SUMMARY

| | | |
|-------|----------------------------------|-----|
| 3.1 | ABOUT THIS CHAPTER | 3-1 |
| 3.1.1 | Conventions | 3-1 |
| 3.2 | COMMAND SUMMARY | 3-1 |
| 3.2.1 | Data Monitor Commands | 3-2 |
| 3.2.2 | File Management Utility Commands | 3-2 |
| 3.2.3 | Floating Point Commands | 3-3 |
| 3.2.4 | Software Breakpoint Commands | 3-4 |
| 3.2.5 | Symbol Processing Commands | 3-4 |

| | | |
|-------|-----------------------------|-----|
| 3.2.6 | Viewport Processing Command | 3-5 |
| 3.2.7 | Command Processing Utility | 3-6 |

4 COMMAND DESCRIPTION AND EXAMPLES

| | | |
|------------|--|-----------------------|
| 4.1 | ABOUT THIS CHAPTER | 4-1 |
| 4.1.1 | Set Software Breakpoint | - BREakpoint 4-2 |
| 4.1.2 | Clear Software Breakpoint | - CLEar 4-3 |
| 4.1.3 | Create Symbol | - CReate 4-5 |
| 4.1.4 | Delete Symbol | - DElete 4-6 |
| 4.1.5 | Temporary Exit to DOS | - DOS 4-7 |
| 4.1.6 | Double Floating Point | - DOUble 4-8 |
| 4.1.7 | Download File | - DOWnload 4-10 |
| 4.1.8 | Single Floating Point | - FLoat 4-12 |
| 4.1.9 | USD-III Command Help | - HElp 4-13 |
| 4.1.10 | Command History | - HIStory 4-14 |
| 4.1.11 | Execute Command File | - INClude 4-15 |
| 4.1.11.1 | Substitute for <Esc> in Command File | - <\> 4-17 |
| 4.1.11.2 | Variables to be Replaced | - %0 - %9 4-18 |
| 4.1.11.3 | Ring the Bell | - BELL 4-19 |
| 4.1.11.4 | Command Display Control | - ECHO 4-20 |
| 4.1.11.5 | Abort/Continue Execution in Case of Error | - ERROR 4-21 |
| 4.1.11.6 | Prompt for Keyboard Input | - GET 4-22 |
| 4.1.11.7 | Redirect Flow of Command Execution | - GOTO 4-23 |
| 4.1.11.8 | Conditional Execution | - IF / ELSE 4-24 |
| 4.1.11.9 | Nest Command File | - INClude 4-25 |
| 4.1.11.10 | Label to be Referred to by "GOTO" Sub-Command | - :label 4-26 |
| 4.1.11.11 | Loop Block | - LOOP / LOOPEND 4-27 |
| 4.1.11.12 | Execution Checkpoint | - PAUSE 4-28 |
| 4.1.11.13 | Comment String | - REMark 4-29 |
| 4.1.11.14 | Set Time Delay Between MICE Free-Run and its Subsequent Command | - TICK 4-30 |
| 4.1.12 | Instruction Step | - IStep 4-31 |
| 4.1.13 | Log Output File | - JOurnal 4-32 |
| 4.1.14 | Load Symbol Table | - LOAd 4-33 |
| 4.1.15 | Log Command File | - LOG 4-34 |
| 4.1.16 | Directory Module | - MODULE 4-35 |
| 4.1.17 | Set Monitor Variable | - MONitor 4-36 |

| | | | |
|--------|---|-----------|------|
| 4.1.18 | Clear Monitor Variable | - NOMon | 4-37 |
| 4.1.19 | Optional Features | - OPTions | 4-38 |
| 4.1.20 | Symbol/Address Query | - QUEry | 4-39 |
| 4.1.21 | Quit USD-III | - QUIT | 4-40 |
| 4.1.22 | Define SoftKey | - SOFTkey | 4-41 |
| 4.1.23 | Include User Symbols from File | - SRecall | 4-43 |
| 4.1.24 | Save Symbols to File | - SSave | 4-44 |
| 4.1.25 | List Symbol | - SYMbol | 4-45 |
| 4.1.26 | Enable/Disable Viewport Update | - UPDate | 4-46 |
| 4.1.27 | Upload File | - UPload | 4-47 |
| 4.1.28 | Recall Window Status and Softkey Definitions | - URECall | 4-49 |
| 4.1.29 | Save Window Status and Softkey Definitions | - USAVe | 4-50 |

5 USD-III MENU COMMANDS

| | | | |
|---------|-----------------------------------|-----------------|------|
| 5.1 | ABOUT THIS CHAPTER | | 5-1 |
| 5.2 | EXECUTING COMMANDS FROM MENUS | | 5-1 |
| 5.2.1 | "File" Pull Down Menu Commands | | 5-3 |
| 5.2.1.1 | Download File | - Download | 5-4 |
| 5.2.1.2 | Upload File | - Upload | 5-7 |
| 5.2.1.3 | Execute Command File | - Include | 5-10 |
| 5.2.1.4 | Log Commands | - Log File | 5-12 |
| 5.2.1.5 | Log Output and Create Output File | - Journal File | 5-14 |
| 5.2.1.6 | Close Log/Journal File | - Close Log/Jou | 5-16 |
| 5.2.1.7 | Quit USD-III | - Quit | 5-17 |
| 5.2.2 | "Setup" Pull Down Menu Commands | | 5-18 |
| 5.2.2.1 | Memory Map | - Map | 5-19 |
| 5.2.3 | "Memory" Pull-Down Menu Commands | | 5-21 |
| 5.2.3.1 | Memory Checksum | - Checksum | 5-22 |
| 5.2.3.2 | Memory Compare | - Compare | 5-23 |
| 5.2.3.3 | Memory Copy | - Copy | 5-24 |
| 5.2.3.4 | Memory Disassembly | - Disassembly | 5-25 |
| 5.2.3.5 | Memory Dump | - Dump | 5-26 |
| 5.2.3.6 | Memory Fill | - Fill | 5-27 |
| 5.2.3.7 | Memory Search | - Search | 5-28 |
| 5.2.3.8 | Memory Test | - Test | 5-29 |
| 5.2.3.9 | Set/Clear Monitor Variable | - Data Monitor | 5-30 |

| | | |
|--------------|---|--|
| 5.2.4 | "Emulation" Pull-Down Menu Commands | 5-34 |
| 5.2.5 | "Trace" Pull-Down Menu Commands | 5-36 |
| 5.2.5.1 | Set/Clear Software Breakpoint | - S/W Breakpoint 5-37 |
| 5.2.5.2 | Break on Read Before Write | - Break on Read 5-39 |
| 5.2.5.3 | Display/Set/Clear Breakpoints | - Event 5-43 |
| 5.2.5.4 | Set Trigger Level | - Trigger Level 5-44 |
| 5.2.5.5 | Trace Control Settings | - Trace Control (for LAM-II) 5-48 |
| 5.2.5.6 | Trace Control Settings | - Trace Control 5-51 |
| 5.2.5.7 | Trace with Backward Count | - Backward Trace 5-55 |
| 5.2.5.8 | Trace with Forward Count | - Forward Trace 5-56 |
| 5.2.5.9 | List Trace Buffer | - List 5-57 |
| 5.2.6 | "Symbol" Pull-Down Menu Commands | 5-59 |
| 5.2.6.1 | Load Symbol Table | - Load 5-60 |
| 5.2.6.2 | Create Symbol | - Create 5-61 |
| 5.2.6.3 | Delete Symbol | - Delete 5-62 |
| 5.2.6.4 | Save Symbols to File | - Ssave 5-63 |
| 5.2.6.5 | Recall Symbols from File | - SRecall 5-64 |
| 5.2.6.6 | Query for Symbol Name | - Symbol Query 5-66 |
| 5.2.6.7 | Query for Address | - Address Query 5-67 |
| 5.2.7 | "Options" Pull-Down Menu Commands | 5-68 |
| 5.2.7.1 | Enable/Disable Symbol/Log/Journal/ S/W Breakpoint | - Toggle Status 5-69 |
| 5.2.7.2 | Enable/Disable Viewport Update | - Update Viewport 5-71 |
| 5.2.7.3 | Define SoftKey Macro | - Define Softkey 5-73 |
| 5.2.7.4 | Save Current Window Status and Softkey Definitions | - Usav 5-74 |
| 5.2.7.5 | Recall Window Status and Softkey Definitions | - Urecall 5-75 |
| 5.2.7.6 | Change Dialog Boxes Colors | - Dialog Color 5-76 |
| 5.2.8 | "Spa" Pull-Down Menu Command | 5-79 |
| 5.2.8.1 | Module Entry Analysis | - Entry 5-83 |
| 5.2.8.2 | Module Duration Analysis | - Duration 5-85 |
| 5.2.8.3 | Execution Time Analysis | - Time 5-87 |
| 5.2.8.4 | Code-Coverage Test | - Coverage 5-89 |
| 5.2.9 | "Lsa" Menu Bar Command | 5-94 |

APPENDICES

A CONVERTER PROGRAM

| | | |
|------------|---|------------|
| A.1 | INTRODUCTION | A-1 |
| A.2 | STARTING CONVERTER | A-1 |
| A.3 | INPUT FILE FORMATS | A-2 |
| A.3.1 | 2500AD Microtek Symbol Format | A-2 |
| A.3.2 | IAR System Format | A-3 |
| A.3.3 | Intel OMF Format | A-4 |
| A.3.4 | Intermetrics Object Language Format | A-4 |
| A.3.5 | MRI Symbol Format | A-5 |
| A.3.6 | OM/LM Format (SYSROF - SYmbol-information Standard Relocatable Object Format) | A-6 |

B USD-III THROUGH PARALLEL PORT

| | | |
|------------|--|------------|
| B.1 | INTRODUCTION | B-1 |
| B.2 | INTERFACING THROUGH PARALLEL PORT | B-1 |
| B.2.1 | IBM-PC/XT/AT MS-PCE Parallel Interface Card Installation and Setup | B-2 |
| B.2.2 | IBM PS/2 MS-MCE Parallel Interface Card Installation and Setup | B-5 |
| B.2.3 | NEC PC-9801 MS-JPC Parallel Interface Card Installation and Setup | B-6 |

C USD-III LIMITED WARRANTY

C-1

INDEX

Index-1

DIRECTORY OF FIGURES

| | | |
|---------------|---|------|
| 1-1 | USD-III System Overview | 1-1 |
| 1-2 | USD-III Screen Layout | 1-5 |
| 1-3 | Typical USD-III Pull-Down Menu Execution | 1-7 |
| 1-4 | Available USD-III Pull-Down Menus | 1-7 |
| 1-5 | An Example of a Space-Bar Toggle Field | 1-10 |
| 1-6 | Default Screen Message Line | 1-11 |
| 1-7a/b | An Example of a Command Syntax Arguments Message Line | 1-13 |
| 1-8 | Pull-Down Menu Message Line | 1-14 |
| 1-9 | Dialog Box Message Line | 1-15 |
| 1-10 | <F1> "Help" Display under Default Screen | 1-17 |
| 2-3 | USD-III Initialization Display | 2-6 |
| 2-4 | "Abort USD-III" Dialog Box | 2-7 |
| 2-5 | "Quit USD-III" Dialog Box | 2-8 |
| 3-1 | IEEE Format for Binary Representation of Floating Point Numbers | 3-3 |
| 4-1 | "Quit USD-III" Dialog Box | 4-40 |
| 5-1 | "File" Pull-Down Menu | 5-3 |
| 5-2 | "Download File" Dialog Box | 5-4 |
| 5-3a | "Download Mode" Choice List (68000) | 5-5 |
| 5-3b | "Download Mode" Choice List (80286/386) | 5-5 |
| 5-4 | "Upload" Dialog Box | 5-7 |
| 5-5a | "Upload Memory Type" Choice List (5052/152) | 5-8 |
| 5-5b | "Upload Memory Type" Choice List (68000) | 5-8 |
| 5-5c | "Upload Mode" Choice List (68030) | 5-9 |
| 5-6 | "Include" Dialog Box | 5-10 |
| 5-7 | "Log Command File" Dialog Box | 5-12 |
| 5-8 | "Journal File" Dialog Box | 5-14 |
| 5-9 | "Close Log/Journal" Dialog Box | 5-16 |
| 5-10 | "Quit USD-III" Dialog Box | 5-17 |
| 5-11 | "SETUP" Pull-Down Menu | 5-18 |
| 5-12 | "Memory Map" Dialog Box | 5-19 |
| 5-13 | "Memory" Pull-Down Menu | 5-21 |
| 5-14 | "Memory Checksum" Dialog Box | 5-22 |
| 5-15 | "Memory Compare" Dialog Box | 5-23 |
| 5-16 | "Memory Copy" Dialog Box | 5-24 |
| 5-17 | "Memory Disassembly" Dialog Box | 5-25 |
| 5-18 | "Memory Dump" Dialog Box | 5-26 |
| 5-19 | "Memory Fill" Dialog Box | 5-27 |
| 5-20 | "Memory Search" Dialog Box | 5-28 |
| 5-21 | "Memory Test" Dialog Box | 5-29 |
| 5-22 | "Data Monitor" Dialog Box | 5-30 |

| | | |
|------|---|------|
| 5-23 | "Operation" Choice List | 5-31 |
| 5-24 | Data Viewport Display | 5-31 |
| 5-25 | "Type" Choice List | 5-32 |
| 5-26 | "Clear" Choice List | 5-32 |
| 5-27 | "Emulation" Pull-Down Menu | 5-34 |
| 5-28 | "Trace" Pull-Down Menu | 5-36 |
| 5-29 | "S/W Breakpoint" Dialog Box | 5-37 |
| 5-30 | "Breakpoint Address" Choice List | 5-38 |
| 5-31 | "Break On Read" Dialog Box (80C186) | 5-39 |
| 5-32 | "Break On Read" Dialog Box (68K/68020/68030) | 5-40 |
| 5-33 | "Global Operation" Choice List (80C186) | 5-41 |
| 5-34 | Setting Memory Block Range in "Range" Field (68K/68020/68030) | 5-41 |
| 5-35 | "Event Setting" Dialog Box | 5-43 |
| 5-36 | "Trigger Level" Dialog Box | 5-44 |
| 5-37 | "Trigger Level Operation" Choice List | 5-45 |
| 5-38 | "Delay Count" Choice List | 5-46 |
| 5-39 | LAM-II "Trace Control" Dialog Box | 5-48 |
| 5-40 | LAM-II "Cycle Qualify" Choice List | 5-49 |
| 5-41 | LAM-II "Timebase" Choice List | 5-50 |
| 5-42 | "Trace Control" Dialog Box | 5-51 |
| 5-43 | "Event Setting" Choice List | 5-52 |
| 5-44 | "Cycle-count Delay" Choice List | 5-53 |
| 5-45 | "Timebase" Choice List | 5-54 |
| 5-46 | "Backward Trace" Dialog Box | 5-55 |
| 5-47 | "Forward Trace" Dialog Box | 5-56 |
| 5-48 | "List Trace Buffer" Dialog Box | 5-57 |
| 5-49 | "List Format" Choice List | 5-58 |
| 5-50 | "Symbol" Pull-Down Menu | 5-59 |
| 5-51 | "Load Symbol Table" Dialog Box | 5-60 |
| 5-52 | "Create Symbol" Dialog Box | 5-61 |
| 5-53 | "Delete Symbol" Dialog Box | 5-62 |
| 5-54 | "Symbol Save" Dialog Box | 5-63 |
| 5-55 | "Symbol Recall" Dialog Box | 5-64 |
| 5-56 | "Symbol Query" Dialog Box | 5-66 |
| 5-57 | "Address Query" Dialog Box | 5-67 |
| 5-58 | "Options" Pull-Down Menu | 5-68 |
| 5-59 | "Toggle Status" Dialog Box | 5-69 |
| 5-60 | "Update Viewport" Dialog Box | 5-71 |
| 5-61 | "Define Softkey" Dialog Box | 5-73 |

CONTENTS

| | | |
|--------------|---|------|
| 5-62 | "Save Setting" Dialog Box | 5-74 |
| 5-63 | "Recall Setting" Dialog Box | 5-75 |
| 5-64 | "Changing Colors" Dialog Box | 5-76 |
| 5-65 | "Spa" Pull-Down Menu | 5-79 |
| 5-66 | Structure of Sample Program | 5-80 |
| 5-67 | Module Entry Chart | 5-83 |
| 5-68 | Module Duration Chart | 5-85 |
| 5-69 | Execution Time Histogram | 5-87 |
| 5-70 | "Code Coverage" Dialog Box (80C186) | 5-89 |
| 5-71 | Setting Memory Block Range in "Range" Field (80C186) | 5-90 |
| 5-72a | Screen Display of "<F9> List" with ASSEMBLY CODE on Display | 5-91 |
| 5-72b | Screen Display of "<F9> List" with MEMORY DUMP on Display | 5-92 |
| 5-72c | Screen Display of Zoomed ASSEMBLY CODE (80C186) | 5-93 |
| 5-72d | Screen Display of Zoomed MEMORY DUMP (80C186) | 5-93 |
| 5-73 | "Lsa" Menu Bar Menu | 5-94 |
| 5-74 | Signal Specification for Logic State Analysis (68000) | 5-95 |
| 5-75 | Logic State Analysis Waveform Display | 5-96 |
| A-1 | CONV3.EXE Typical Operation | A-1 |
| B-1 | MICE-III Communication interface Connectors | B-1 |
| B-4 | MS-PCE Port Selection Example | B-4 |
| B-5 | MS-PCE Status Port Bit Assignment | B-5 |
| B-5 | MS-JPC DIP Switches U4 & U5 Port Selection Example | B-6 |

DIRECTORY OF TABLES

| | | |
|------------|--|------------|
| 2-1 | Pin Assignment for Serial Port Interface | 2-4 |
| 2-2 | Required Interface Card for Parallel Port Communication | 2-4 |
| A-2 | Applicable Options for Intel Tools | A-4 |
| A-3 | Applicable Options for MRI Tools | A-5 |
| B-2 | IBM-PC/XT/AT MS-PCE Parallel Interface Card Interface Card Installation Communication | B-2 |
| B-3 | Pin Assignment and Signals for Parallel Interface | B-3 |



Chapter 1

INTRODUCTION

MICROTEK INTERNATIONAL INC.

1.1 WHAT IS USD-III ?

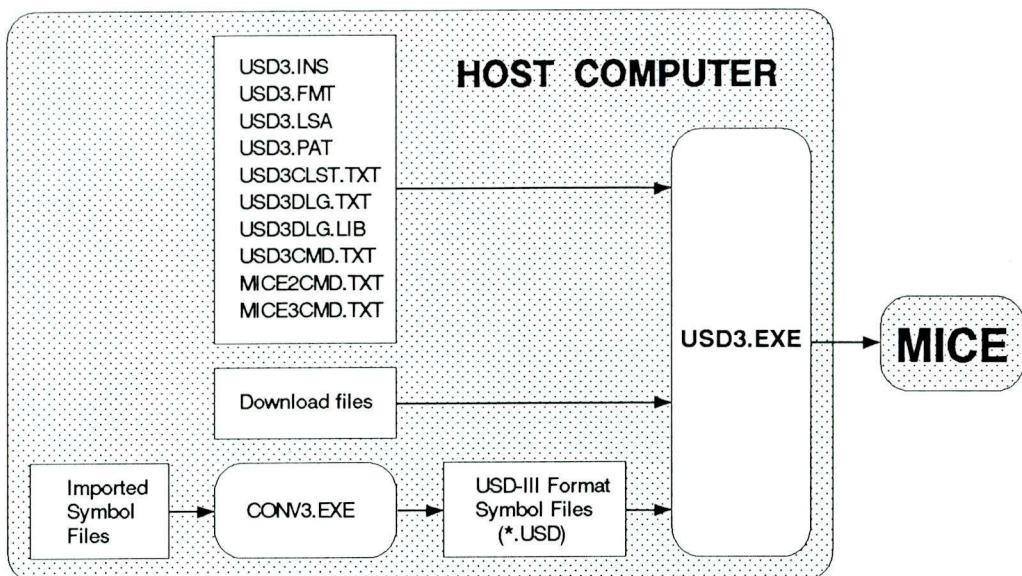


Figure 1-1 USD-III System Overview

USD-III is a text windowed symbolic debugger for MICE-II and MICE-III Series of in-circuit emulators. It communicates with user through MS-DOS environment and interfaces with MICE through either RS-232C or parallel ports. USD-III is enhanced with several new features not previously available with the preceded generation of USD. With USD-III, MICE will be able to extend its inherent emulation capabilities to include the following:

- User Friendly Interface:
 - Command execution from command line.
 - Command execution from pull-down menu an dialog box.
 - Zoomable, scrollable windows (viewports)
 - 8-color selection for both viewport background and text

- Seven Viewports:
 - Command Viewport
 - Code Viewport
 - Register Viewport
 - Breakpoint Viewport
 - Stack Viewport
 - Trace Viewport
 - Data Viewport
- 256 software breakpoints
- Two level on-line help
- Up to 40 characters symbol name
- Floating point command:
 - Single and double precision form
- Enhanced command file feature
- User definable softkeys
- Mouse user interface

1.2 USD-III AT A GLANCE

Host Computer - IBM PC/XT/AT (at least 286 machines), PS/2 or compatibles with CGA, VGA, EGA, MCGA, PGS or Hercules display monitor.
- NEC PC-9801 family.
- Equipped with at least 640Kbyte RAM or above, 1Mbyte of extended memory and a hard disk.

Operating System - MS-DOS Version 2.1 or later.

Communication Interface - RS-232C and parallel (refer to MICE manual for details on parallel interfaces through add-on MS-PCE, MS-MCE and MS-JPC interface cards).

Mouse Support - V2.0 is fully mouse-compatible supporting a two-button mouse function. USD-III features may be accessed with either mouse or keyboard.

Transparency - All models under MICE-II and MICE-III series, are supported.
- All MICE commands are transparent to USD-III.
- Downloads and uploads object code to and from MICE respectively.

Download Format - Intel hex-format
- Intel OMF format
- Motorola S-format
- Tektonix hex-format

Symbol Table Size - Unlimited number of modules and symbols; maximum of 300 user created symbols.

Supported Tool Chain - Intel tool chain

- "InterTools"
- MRI tool chain
- MicroSoft tool + SSI's "LINK & LOCATE++"
- 2500AD assembler
- IAR tool chain
- Phar Lap tool chain

USD-III Files - USD3.EXE

USD-III execution file.

- USD3.INS

Installation and initialization file.

- USD3.FMT

Format file for all MICE. It provides MICE output format information to USD-III.

- READ.ME

Documentation file containing recent release information.

- CONV3.EXE

Converts imported symbol file to USD-III symbol file format. It extracts the symbol information from output files of all the popular cross-assemblers and cross-compilers, and converts them into a uniform symbol file transparent to USD-III. The converter output files are then identified with ".USD" extension. Execution of CONV3.EXE is described in Appendix A of this manual.

- MCE.EXE

MS-MCE port address setting file.

- @60AC.ADF

MS-MCE diagnostic file.

- USD3.LSA

Logic state analysis configuration file.

- USD3.PAT

ASCII character pattern file for graphic function.

- USD3CLST.TXT

Auxiliary file for dialog boxes.

-
- **USD3DLG.TXT**
Help text file for dialog boxes.
 - **USD3CMD.TXT**
Help text file for USD-III commands.
 - **MICE2CMD.TXT**
Help text file for MICE-II commands.
 - **MICE3CMD.TXT**
Help text file for MICE-III commands.
 - **USD3DLG.LIB**
Library for dialog boxes.
 - **USD3DLG.INX**
Index file for help text.

1.3 INTRODUCTION TO MOUSE OPERATION

USD-III Versions 2.0 and latter supports mouse operation. Generally, the mouse left button is used to activate a USD-III function. The right button functions as <ESCAPE> key. Where three-button mouse is used, the center button has no function under USD-III.

The following mouse convention applies throughout this manual.

- Move** move the mouse pointer (cursor) without pressing button.
- Drag** move the mouse pointer while pressing the left button.
- Point** move the mouse pointer over an item, menu, object, etc.
- Click** Press and release button while mouse pointer is on an item, menu, object, etc.

Where a mouse application is explained in this manual, it is preceded with graphic symbol



1.4 INTRODUCTION TO USD-III SCREEN LAYOUT

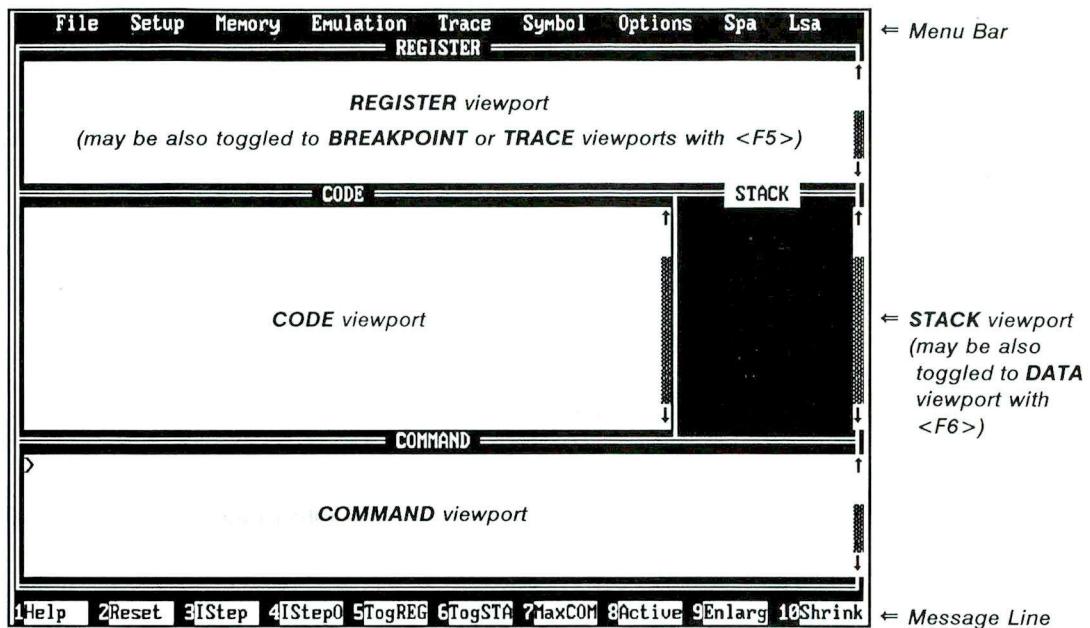


Figure 1-2 USD-III Screen Layout

1.4.1 Menu Bar

The Menu Bar has nine menu items. A sub-menu or pull-down menu may be pulled down from each of the items by holding down the <Alt> key and pressing the key-character (highlighted on monochrome screen or shown in red on EGA/VGA screen) of the item, or as summarized below:

| <u>Keystroke</u> | <u>To Access</u> | <u>Function</u> |
|------------------|------------------|---|
| <Alt-F> | File Menu | File facilities, exit to DOS, version display, etc. |
| <Alt-S> | Setup Menu | Save and recall MICE setup, memory mapping, etc. |
| <Alt-M> | Memory Menu | Memory related operations |
| <Alt-E> | Emulation Menu | Instruction and cycle stepping |
| <Alt-T> | Trace Menu | Trace program execution |
| <Alt-Y> | Symbol Menu | Symbolic debugging information |
| <Alt-O> | Options Menu | Save/recall window status/softkey definitions, etc. |
| <Alt-P> | Spa Menu | Software performance analysis |
| <Alt-L> | Lsa Menu | Logic state analysis |



With mouse, mouse cursor may be moved to the desired menu in the Menu Bar, click the left-button to pop-up the pertinent pull-down menu.

A brief description for each of the menus may be viewed from screen by pressing <F1> twice while the selected menu's pull-down menu is on display.

1.4.2 USD-III Viewports

The USD-III screen (Figure 1-2) is divided into smaller child windows called "viewports". A total of six viewports are available:

- REGISTER** Displays the current CPU registers' content.
- CODE** The codes where current program counter (PC) is located, are displayed in this viewport.
- COMMAND** This is the command line viewport. Commands and their responses are all displayed in this viewport.
- BREAKPOINT** Displays the current software breakpoint information including the breakpoint number and breakpoint address.
- TRACE** Current trace buffer is displayed in this viewport.
- STACK** The current stack information are displayed in this viewport.
- DATA** All monitored variables are displayed in this viewport.

The default screen will only display the REGISTER, CODE and COMMAND viewports. To reconfigure window layout and viewports, use the following keystrokes:

- <**F5**> Toggle REGISTER viewport to BREAKPOINT or TRACE
- <**F6**> Pop-up/ pop-down STACK or DATA viewport when needed
- <**F7**> Zoom COMMAND viewport from default to maximum size and vice-versa
- <**F9**> Zoom-up (enlarge) active viewport (highlighted or indicated by a different color) to desired size
- <**F10**> Zoom-down (shrink) active viewport to desired size
- <**Ctrl-F8**> Enter active viewport into scrolling mode (indicated by "↑↓")

<Ctrl-F9> Change active viewport background color

<Ctrl-F10> Change active viewport text color

Reconfigured window status may be saved with "USAve" command (Section 4.1.24). Saved status becomes the default screen the next time USD-III is invoked.

Except for COMMAND, all other viewports display the most current status of information as a result to a recently executed command. Because the real-time updating of viewports affects command execution speed, viewport updating may be disabled with the "UPDate" command (Section 4.1.21).

1.4.3 Pull-Down Menu

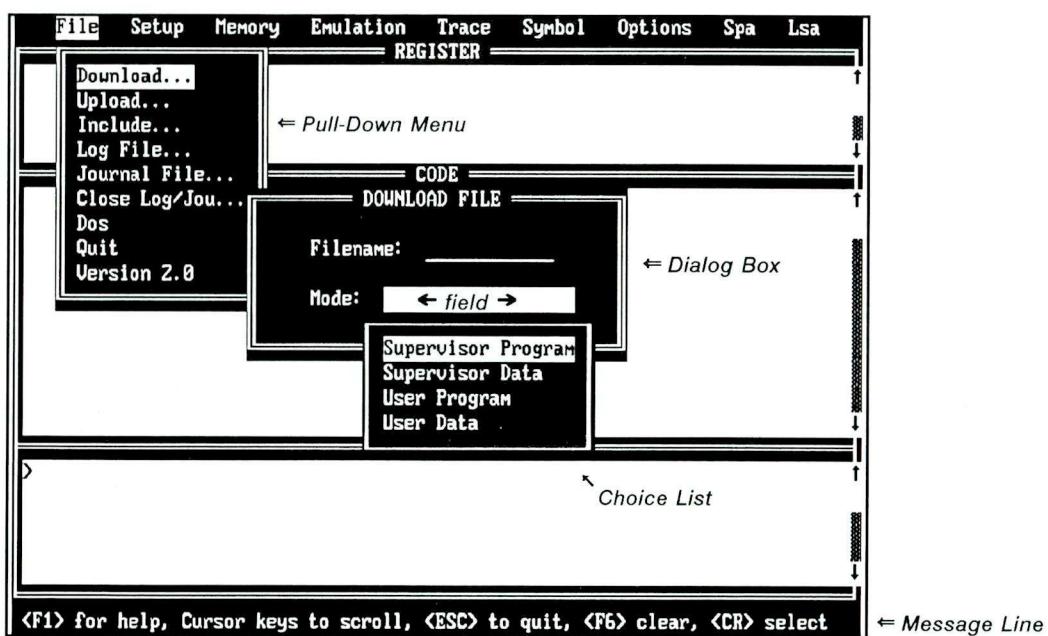


Figure 1-3 Typical USD-III Pull Down Menu Execution

USD-III pull-down menu (Figure 1-3) displays the group of menu commands under each of the menu bar items. USD-III has 9 pull-down menus as illustrated in Figure 1-4.

INTRODUCTION

<↑><↓> may be used to move and highlight the desired command from the pull-down menu, then press <CR> to execute it. The other way to execute the command is to directly key-in the command key character (highlighted on monochrome screen or shown in red on EGA/VGA).

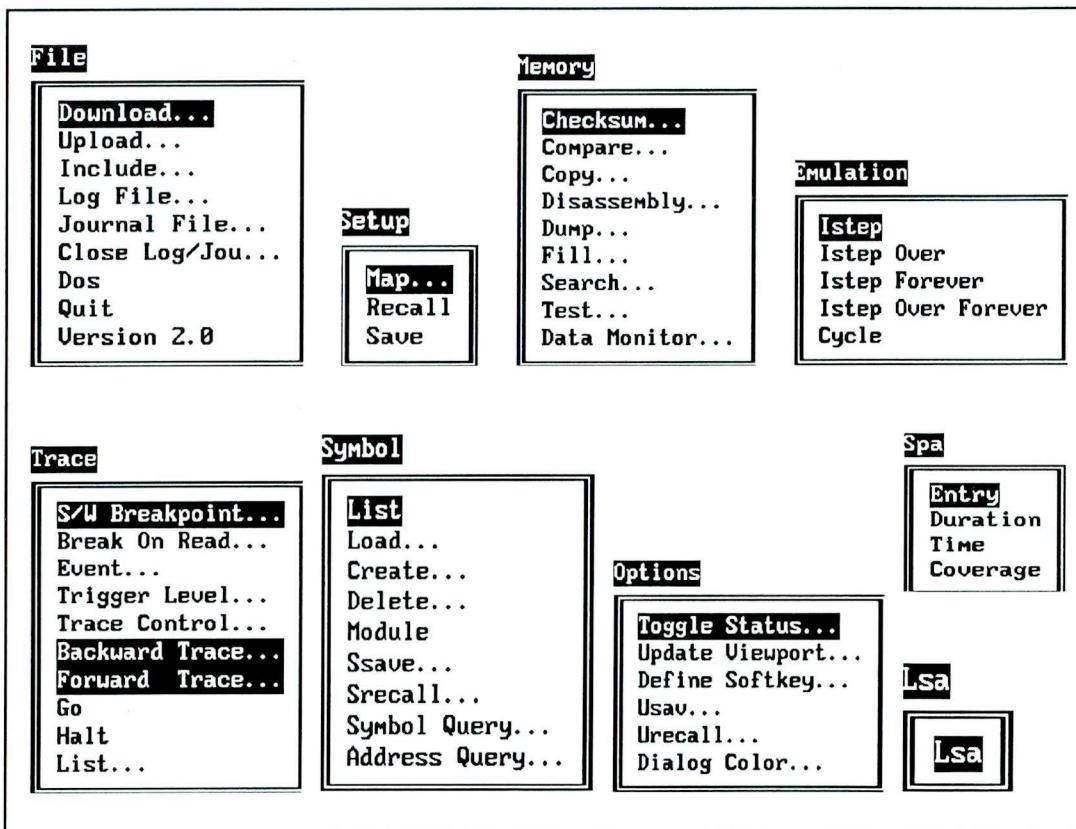


Figure 1-4 Available USD-III Pull-Down Menus



Point the mouse cursor on the desired command in the menu, and click the left-button to execute.

1.4.4 Pop-Up Dialog Box and Choice List

Dialog box (Figure 1-3) will display whenever pull-down menu commands with ellipses (three dots) suffix is initiated. It prompts user for the needed parameters to carry out a command. There are three methods in which needed parameter are entered into active fields. A dialog box may contain a field or several fields.

1. A field may be a simple field where a data is directly keyed-in as in the case of "Filename" field in the "DOWNLOAD FILE" dialog box shown in the above figure. Some fields limit the length of data that could be entered, but most of the fields could accept entries longer than the length of the fields provided. Once the field is saturated, the left end characters are pushed out from the field (hidden). Use **<↔>** to view the hidden portion of the string.
2. A field may provide a choice list which is retrievable by pressing **<F2>**. When **<F2>** is required to retrieve a choice list, the message "**<F2> to show choices**" will display in the Message Line. The choice list provided by the "Mode" field in the above figure was retrieved with **<F2>**.

With choice list on display, use **<↑><↓>** to select an appropriate item from the list and press **<CR>**. Where the listing is too long to fit in one display box, a vertical scroll bar will prompt user to scroll through the rest of the listing. The selection is entered into the highlighted field and the choice list quits at the same time.

3. A field may have only two options to choose from, e.g., ON/ OFF; ADD/ DELETE (as in the case of "Operation" field shown in Figure 1-5); YES/ NO; etc. In this case, USD-III entered one of the alternatives into the field by default. Move cursor on the field and press **<space-bar>** to toggle between the two choices. When toggle selection is required, the message "**<SPACE> to toggle the options**" will display in the Message Line.

By pressing **<F1>** with cursor placed on any of the fields, USD-III "Help" messages will pop-up to provide an instant explanation on what or how to fill the field.

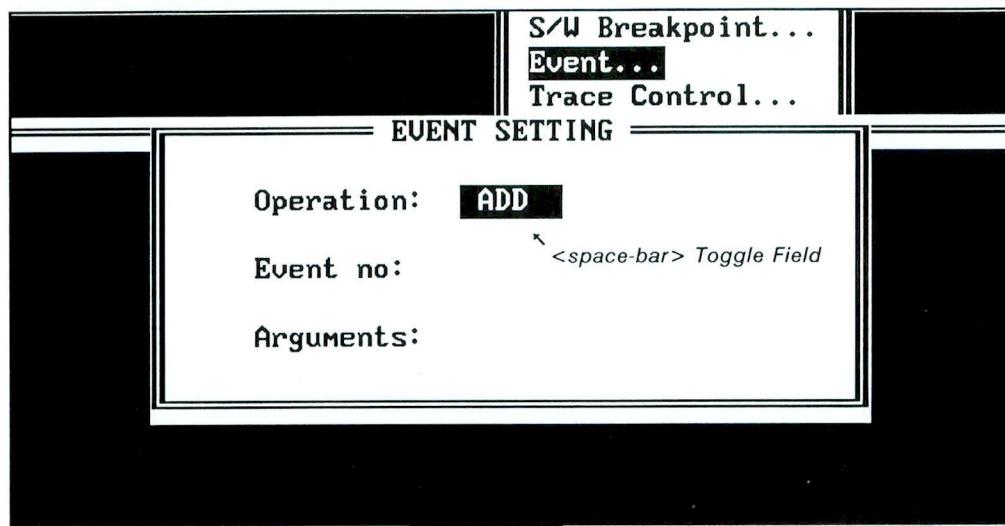
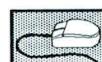


Figure 1-5 An Example of a Space-Bar Toggle Field



Use the following guide to move around dialog box with mouse.

| move mouse cursor to- | click left-button to- | click right-button to- |
|---|---|------------------------|
| any area in the dialog box other than the field. | execute dialog box setting | |
| the active field which provides choice list and <space toggle> selection. | -pop-up a choice list. -toggle between two options, e.g., ON/OFF; YES/NO ADD/DELETE, etc., | quit dialog box |
| inactive field. | switch field to active in order to fill in required data | |

1.4.5 Message Line

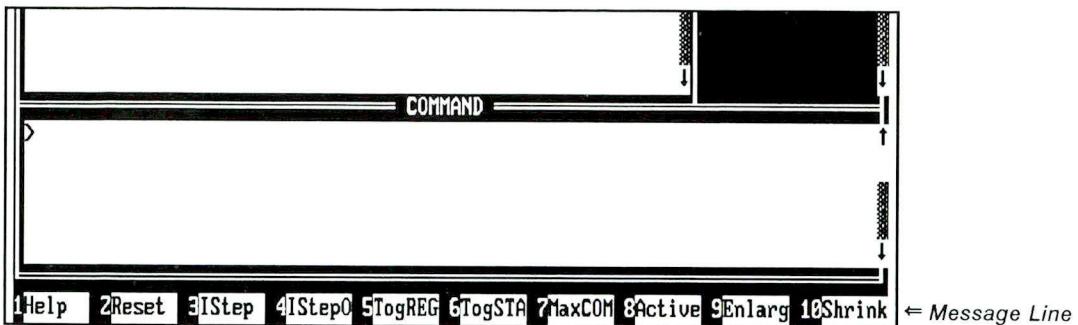


Figure 1-6 Default Screen Message Line

The Message Line provides an instant on-screen clue to frequently used keystrokes for USD-III executions. These keystrokes are directly applicable to the USD-III screen status on display. Therefore, Message Line messages varies as the screen display changes as summarized in the next sections.

1.4.5.1 Default Screen Message Line

With default screen, Message Line (Figure 1-6) will display a set of function keys (F1 - F10) functions as described below. The functions described are effective only for as long as they are on display or until user changes the screen status by keying command in COMMAND viewport or calling in pull-down menu, dialog box, choice list, etc. Notice that Message Line messages changes to another set of keystroke functions at every change of screen status.

- <F1> On-line help (displays USD-III command syntax and function keys definitions)
- <F2> MICE-II "r" command or MICE-III "RESet" command
- <F3> MICE "IStep" command
- <F4> MICE "IStep Over" command
- <F5> Switch BREAKPOINT viewport to REGISTER, then to TRACE viewports, and back to BREAKPOINT viewport.
- <F6> Toggle STACK or DATA viewport ON/ OFF
- <F7> Zoom COMMAND viewport from default to maximum size and vice-versa.

INTRODUCTION

- <F8> Select "active viewport" (active viewport is indicated by change in viewport title color)
- <F9> Zoom up (enlarge) active viewport size at one line increment.
- <F10> Zoom down (shrink) active viewport size at one line decrement.



Point mouse cursor on the desired Message Line menu and click to execute.

In addition to the above function keys, the following functional keystrokes (not displayed on Message Line) also applies to the default screen. These are:

- <Ctrl-F8> Enter active viewport into scrolling mode ("↑↓" will display to indicate scrolling mode)
- <Ctrl-F9> Change active viewport background color
- <Ctrl-F10> Change active viewport text color
- <Ctrl-PgUp>
- <Ctrl-PgDn> Retrieve previous/next buffer data of BREAKPOINT/TRACE viewports (available only under scrolling mode)
- <Ctrl-C> or
- <Ctrl-Break> Abort USD-III
 - <↑><↓> Retrieve last/next history buffer command
 - <Alt-?> Pull-down menu selection (?=highlighted key character)
- <Shift-F1>...
- <Shift-F10> Execute softkey K1...K10

All of the above may be viewed on screen by pressing <F1> "Help".

1.4.5.2 Message Lines for Current Command

As soon as the first character of a command is entered in the COMMAND viewport, all the available USD-III and MICE command keywords having the same initial character with the one being entered, will also display in the Message Line. Figure 1-7a below shows the Message Line displaying all command keywords starting with "C".

Right after the full command keyword is keyed into command line, the Message Line changes to display all the corresponding parameters or arguments of the command as shown in Figure 1-7b. Like the command keyword, the parameters has to be keyed into the command line.

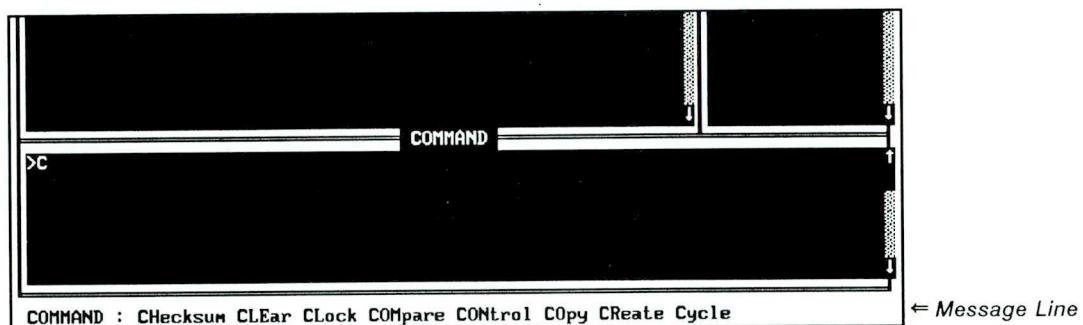


Figure 1-7a An Example of a Command Syntax Keywords Message Line

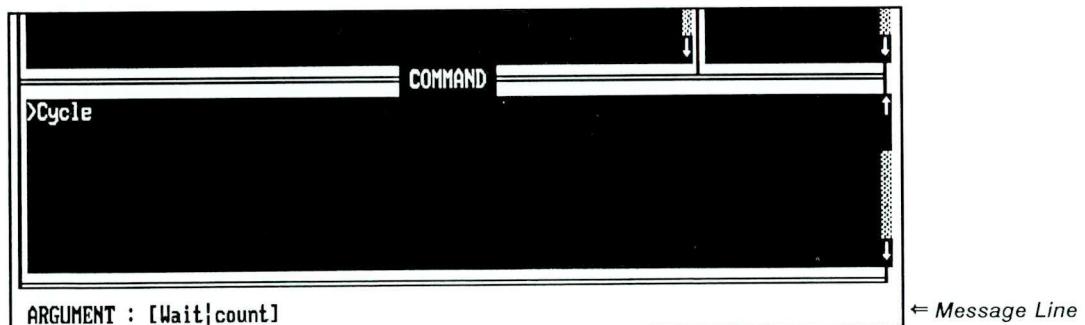


Figure 1-7b An Example of a Command Syntax Arguments Message Line

1.4.5.3 Message Line Status with Pull-Down Menu on Display

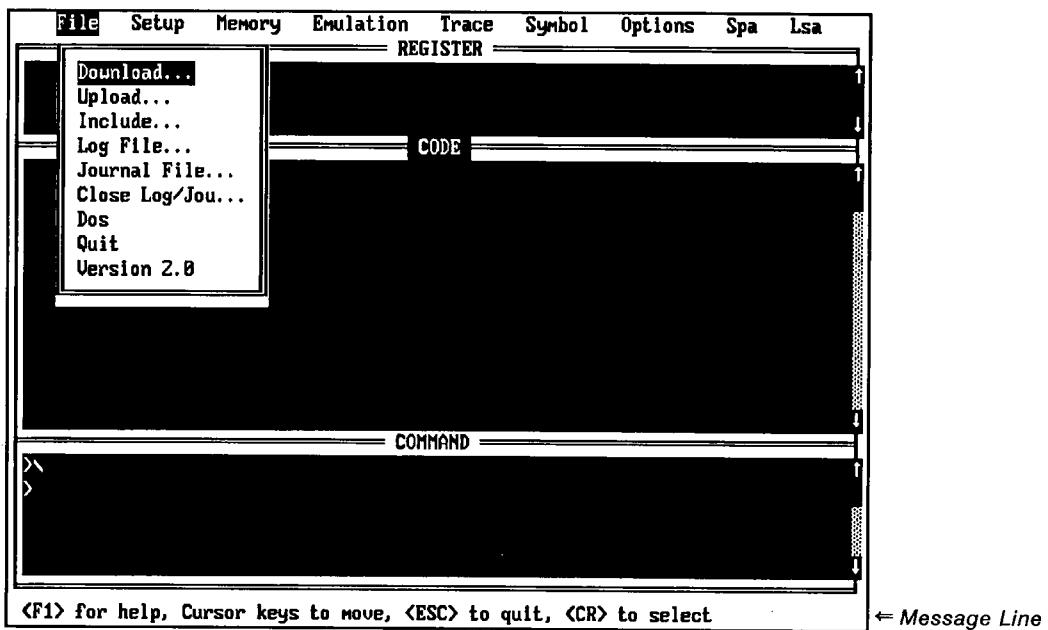


Figure 1-8 Pull-Down Menu Message Line

Pull-down menu Message Line (Figure 1-8) provides hints to following keystrokes:

<F1> On-line "Help" describing the function of the command

Whenever a pull-down menu appears on screen, mouse can NOT point and click any of the Message Line menus.

<>><<>

<↑><↓> Scroll to select commands from the menu

<Esc> Quit current pull-down menu

<CR> Execute the selected command. Another way to execute the command is to directly key-in the highlighted command key-character.

1.4.5.4 Message Line Status with Dialog Box on Display

The following keystrokes are defined in the dialog box Message Line (Figure 1-9):

- <F1> On-line "Help" describes the purpose and what to enter into the field on cursor
- <Esc> Quit the current dialog box without executing any command nor saving any value
- <F2> Call-in choice list. Will display and available only for fields that provide choice list.
- <F6> Clear the contents of a field
- <F10> Execute command after all parameters are set.

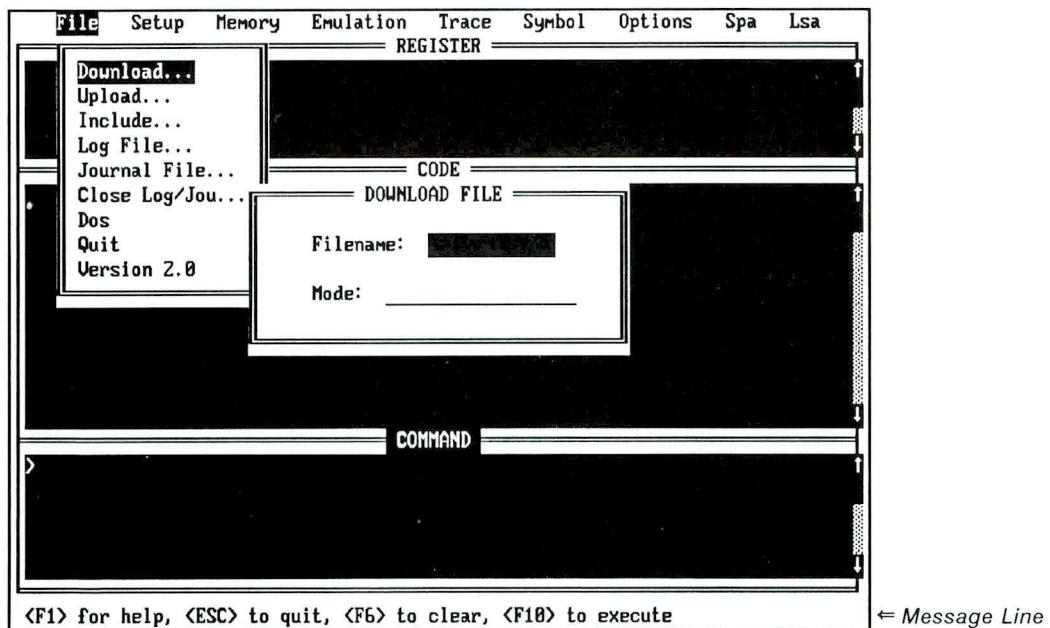


Figure 1-9 Dialog Box Message Line

In addition to the ones on Message Line display, the following keystrokes are also applicable when editing the fields in a dialog box:

- <CR> Moves cursor to the next field down the line. When press repeatedly until cursor leaves the last field, it saves (with most dialog boxes), the field settings and exits from dialog box at the same time.

| | |
|--------------------|---|
| <Shift-F3> | Undo a field, i.e. redisplay the previous value |
| <Ctrl-Home> | Go to the first field in the dialog box |
| <Ctrl-End> | Go to the last field in the dialog box |
| <Home> | Go to the beginning of the field |
| <End> | Go to the end of the field |
| <↑> or <Shift-Tab> | Go to the previous field |
| <↓> or <Tab> | Go to the next field |
| <↔> | Move cursor one character left |
| <→> | Move cursor one character right |
| <Insert> | insert-toggle |
| <Delete> | delete a character |
| <Backspace> | backspace, delete a character backward |
| <Space-bar> | toggle between two options. |

1.4.5.5 Message Line Status with Choice List on Display

The following keystrokes definitions are displayed in the Message Line (Figure 1-3) when the screen displays a choice list:

| | |
|-------|---|
| <F1> | On-line "Help" describes the purpose and what to enter into the highlighted field in connection with the choice list on display |
| <↑> | Move cursor up one choice list item |
| <↓> | Move cursor down one choice list item |
| <Esc> | Quit the choice list |
| <F6> | Clear the current field and quit the choice list |
| <CR> | Enter selection and exit the choice list |

Though not displayed in the Message Line, the following keys also work when choice list is on screen:

| | |
|--------|----------------------------------|
| <Home> | Go to the first item on the list |
| <End> | Go to the last item on the list |

1.5 ON-LINE <F1> "HELP" SUMMARY

The on-line <F1> "Help" will provide different help information when invoked at different screen status. Here is a summary of what help information is to expect from what screen display:

a) Default screen (Figure 1-2):

Under this screen status, pressing <F1> will provide a full screen listing of function keys definitions and a summary of USD-III command syntax (Figure 1-10). Use <↑><↓> or <PgUp><PgDn> to scroll through the two-page listing.

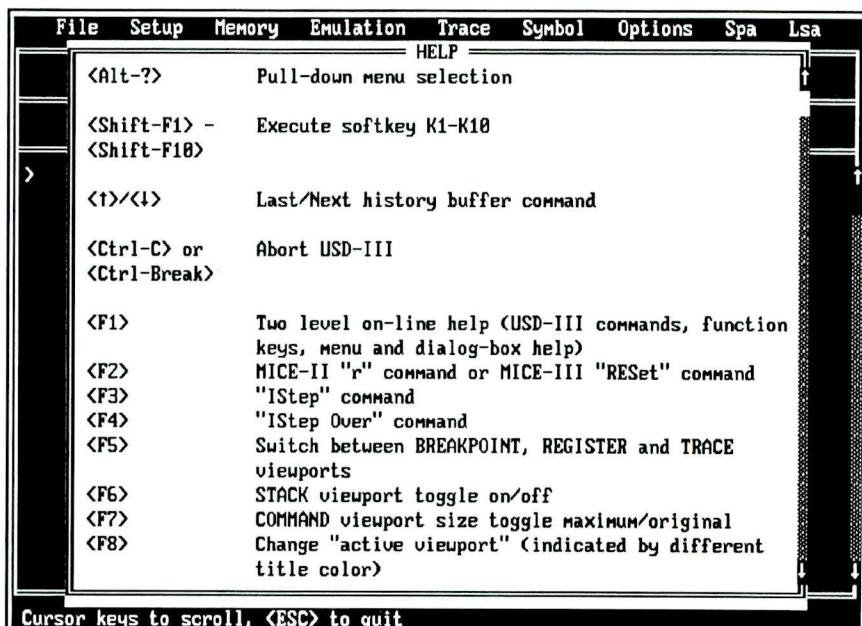


Figure 1-10 <F1> "Help" Display under Default Screen

 Mouse could only point and click "Help" menu from default screen Message Line. Once in the Help window, point mouse cursor on the up-arrow at right top corner of the window and press left-button to scroll up. Likewise, point to the bottom right arrow to scroll down.

b) Screen with pull-down menu on display (Figure 1-8):

With the cursor on one of the pull-down menu commands, pressing <F1> will provide a feature description of the current commands, i.e., the command in which the cursor is highlighting.

Pressing <F1> a second time while the first "Help" information is still on display, will provide a descriptive information of the current pull-down menu.

c) Screen with dialog box and/or choice list on display (Figure 1-9):

Pressing <F1> while the cursor is on a field in a dialog box will provide a "Help" display providing hints on what to enter in the field.

Pressing <F1> a second time will also provide the same "Help" feature description of the current command as discussed in Item b) above.

Press <Esc> to exit from Help window.



Click right-button to quit from Help window.

Chapter 2

GETTING STARTED

MICROTEK INTERNATIONAL INC.

2.1 WHAT YOU SHOULD GET

The standard USD-III package includes the following items:

- One (1) 1.2M-byte floppy disk containing USD-III files and installation program.

NOTE

Make a backup copy of the disk.

- A copy of USD-III User's Guide
- A warranty card

For complete listing of the items, refer to the shipping package checklist.

2.2 INSTALLATION AND BACKUP PROCEDURE

It is advisable to have at least one backup copy of the package stored in a safe place. Use MS-DOS "DISKCOPY" utility for this purpose.

The USD-III package includes an installation program which will guide user thought the step-by-step installation process or as described below:

1. Insert the diskette labeled "USD-III" into drive A.
2. Key-in >a:install<CR>
User will be asked to provide destination drive and directory.
3. Append the current search path to include the **USD3** directory. Use the DOS PATH and APPEND commands to append command and data-file search to the **USD3** directory.
4. Modify the system configuration file CONFIG.SYS as follows:
FILES = 20
BUFFERS = 30

5. Reboot the system after modifying CONFIG.SYS file.

After completing the installation procedure, compare the files in **USD3** directory with the ones listed in Section 1.2.

2.3 USD-III SETUP ON MS-DOS/PC-DOS HOSTS

NOTE

Check to ensure that the host computer is properly linked with MICE in accordance with the proper procedures and interface parameters described in the pertinent MICE User's Manual.

USD-III defaults at installation are as follows:

- COM1 serial port
- 19200 baud rate
- 8 data-bit
- None parity
- Handshake code 03(ETX).

User may change the existing setup status in the "USD3.INS" file by using the formats options described below:

1. Vector address selection for Software Breakpoints service routine:

- **V:address**
- **V:N** (Disable Software Breakpoint)

The default vector address is determined by USD3.FMT file.

2. Specification for auto-executing command file after USD-III is loaded:

- **I:filename**

3. Options to stop or keep MICE's free-running after USD-III is loaded:

- **R:y** Keeps MICE in free-running status. Note that under this option, nothing may be displayed in the REGISTER, and CODE viewports.
- **R:n** Stops MICE free-running status when USD-III is loaded.

If this option is not specified, USD-III will choose "**R:n**" by default.

4. Serial port status setting; COM1 or COM2, Baud-rate (150, 300, 600, 1200, 2400, 4800, 9600, 19200), Data-bit (7,8) and Parity (N,0,E):
 - S:COM1, 19200, 8, N
5. Parallel port setting:
 - P:200
6. MICE handshake code:
 - H:03 With MICE-III series, user may specify its own handshake code which will overwrite whatever is the existing MICE handshake code.

For example, if the user's PC is connected to the MICE at serial port COM1 at 9600 baud-rate, 7 data-bit, even parity and handshake code 03, revise the default values as follows:

```
COPY CON USD3.INS
S: COM1,9600,7,E
H: 03
<Ctrl-Z>
```

2.3.1 RS-232C Serial Port Setup

Communication between the host computer and MICE is accomplished by linking an RS-232C cable between the host computer's RS-232C serial port and the MICE RS-232C output port. Be sure that the baud rate and other parameters are properly set for both in accordance to their respective instruction. The RS-232C cable connector pin configuration is illustrated below.

NOTE

Any pin not indicated below are not to be connected. User must correctly determine the right type of interface requirements for their controlling device.

| MICE | IBM-PC/XT IBM-PS/2 NEC PC-9801 | IBM-PC/AT | Pin Definition |
|---------------|--------------------------------------|-----------|---------------------|
| Pin 2<----->2 | | 3 | Transmit Data |
| 3<----->3 | | 2 | Receive Data |
| 4<----->4 | | 7 | Request To Send |
| 5<----->5 | | 8 | Clear To Send |
| 6<----->6 | | 6 | Data Set Ready |
| 7<----->7 | | 5 | Signal Ground |
| 20<----->20 | | 4 | Data Terminal Ready |

Table 2-1 Pin Assignment for Serial Port Interface

2.3.2 Parallel Port Setup (MICE-III only)

In addition to Serial/Serial module (SSM), MICE-III also provides a Parallel/Serial Module (PSM). This module, when needed, may be attached to replace the SSM on the Control Processor Module (CPM) board.

With PSM, the MICE can communicate with a host computer through either parallel or serial interface ports. Note that communication through parallel port is supported on NEC PC-9801, IBM PS/2, IBM PC/XT/AT or compatible only and requires an interface card (provided with the MICE package) to be installed in the host computer. Three types of interface card are currently available:

| HOST COMPUTER | IBM PC/XT/AT or compatible | IBM PS/2* | NEC PC-9801 |
|----------------|-------------------------------|-----------|-------------|
| INTERFACE CARD | MS-PCE** | MS-MCE | MS-JPC |

* Applies to Models 50, 60, 70 and 80 only. For Model 30, MS-PCE is used.
 ** Unless otherwise specified, this board (MS-PCE) is shipped with MICE.

Table 2-2 Required Interface Card for Parallel Port Communication

When using parallel port, user must define the following parallel port address in USD3.INS file.

P: port address

The base address of I/O ports is the port address on MS-PCE and MS-MCE cards, or the data port address on the MS-JPC card. The address of status port is the address, next and adjacent to data port address.

The use of the serial and parallel interface are mutually exclusive. If both P: option and S: option are specified in USD3.INS file, the S: option will be ignored.

Note that the default I/O ports address setting of the switches on MS-PCE and MS-MCE is at 200H, and the default MS-JPC setting are; data port at 0C0H, and the status port is at 0C1H (the addresses of the data port and the status must be consecutively adjacent). Refer to Appendix B for more details on parallel port setup and operation.

MICE-III versions supporting the parallel interface are:

- | | | | |
|-----------------|---------------|------------------|---------------|
| ■ MICE-32 68020 | V3.0 or later | ■ MICE-16 8086F | V3.0 or later |
| ■ MICE-32 68030 | V3.0 or later | ■ MICE-16 80286 | V3.0 or later |
| ■ MICE-32 80386 | V3.0 or later | ■ MICE-16 80C186 | V3.0 or later |
| ■ MICE-16 H8 | V3.0 or later | ■ MICE-8 64180 | V1.1 or later |
| ■ MICE-16 V25 | V3.0 or later | ■ MICE-8 Z80 | V2.3 or later |
| ■ MICE-16 68000 | V3.0 or later | | |

2.4 INVOKING USD-III

Start USD-III by keying- **>USD3<CR>**

The Initial USD-III screen will display current USD-III released version and the communication link information.

If USD-III is linked with MICE-II, MICE self-test process will start immediately. If linked with MICE-III, enter-

M<CR>

to initiate MICE self-test. Observe self-test messages appear on the initial screen (Figure 2-3). If the linking process is successful, "***Linking MICE successfully, please wait...***" message will display at the bottom of the screen. The screen is then switched to the USD-III default window status and screen layout (shown in Figure 1-2).

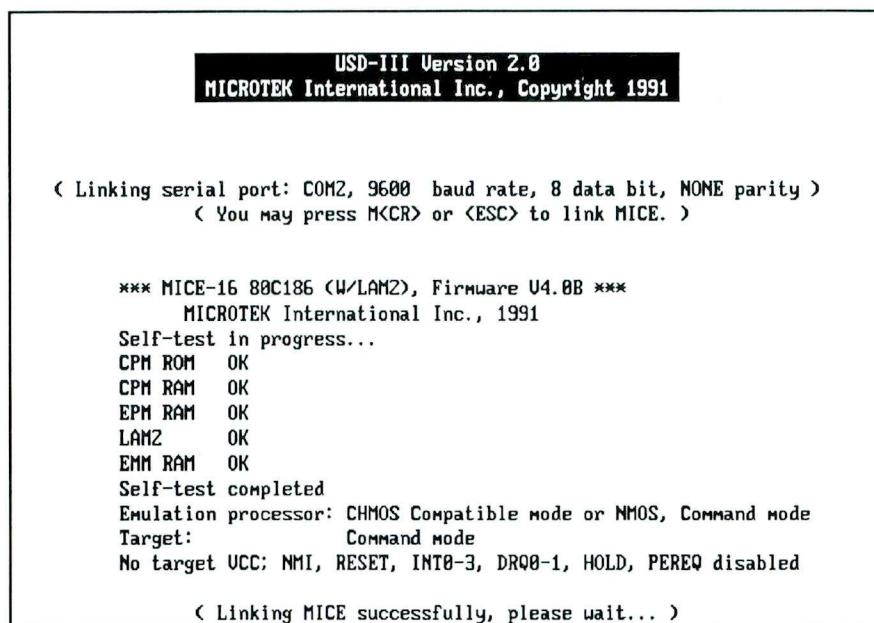


Figure 2-3 USD-III Initialization Display

If an error is detected, the screen display freezes. User may press <Ctrl-C> or <Ctrl-Break> to abort and quit USD-III. Resulting screen is illustrated in Figure 2-4 (next page).

2.5 QUITTING USD-III

There are two methods of exiting from USD-III:

1. By executing "QUIT" command

This command may be executed by keying the command directly into the COMMAND viewport (Section 4.1.19) or through the "FILE" pull-down menu command (Section 5.2.1). A dialog-box will display to confirm whether user wishes to "**Save screen and softkey ?**" (Figure 2-5) after executing the command. If the answer is "Y", USD-III will execute the "USAve" command (Section 4.1.25/ 5.2.7.4) internally before exiting. If the answer is "N", the system will immediately exit to MS-DOS.



Click left-button with mouse cursor pointing on "FILE" menu of Menu Bar. With "FILE" pull-down menu on display, point "QUIT" command and click left-button again to execute.

2. By pressing <Ctrl-C> or <Ctrl-Break>

With this method, USD-III is abruptly aborted from anywhere within USD-III. The message "**Abort USD-III? Y**" (Figure 2-4) will display whenever this method is used.

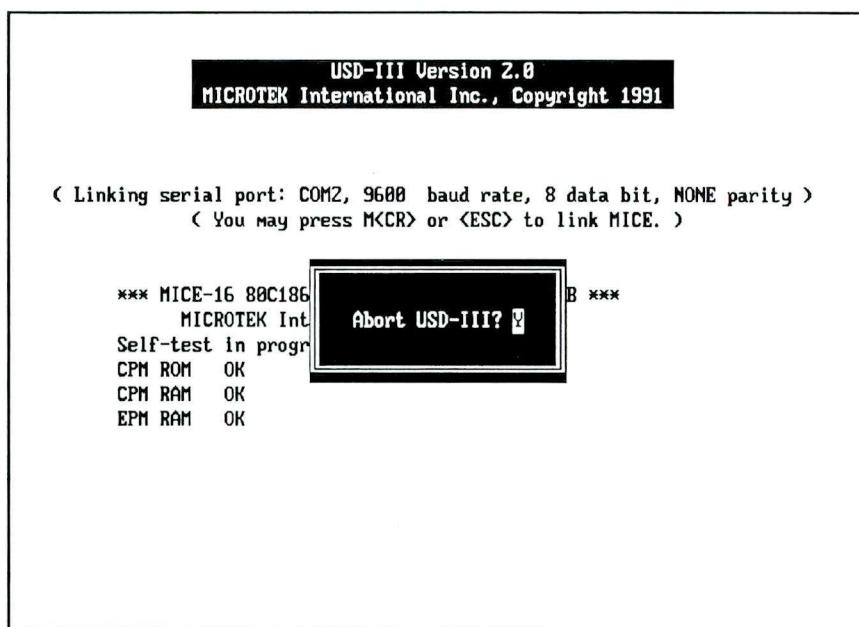


Figure 2-4 "Abort USD-III" Dialog Box



To abruptly abort USD-III, Mouse can only be applied after the above dialog box appears. Click left-button (disregard cursor position) to confirm "Yes" response and display the second exit dialog box (Figure 2-6), or click right-button to return to USD-III. With the second dialog box on display, click left-button for "No" or right-button for "Yes" response.

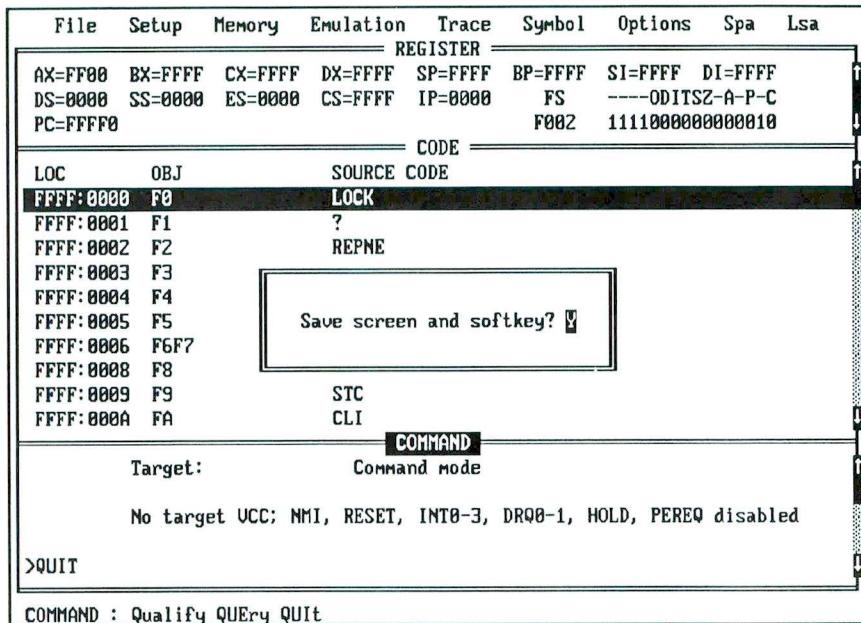


Figure 2-5 "Quit USD-III" Dialog Box

2.6 SOFTWARE BREAKPOINT IMPLEMENTATION

USD-III implements software breakpoints by substituting original code in the desired address of the user program, with a trap code (for example, "INT 3" code for 80x86 CPU series, "ILLEGAL INSTRUCTION" for 680x0 CPU series). It then sets a hardware breakpoint, i.e., Event 5 for MICE-III with LAM, Event 7 for MICE-III with LAM-II or H3 breakpoint for MICE-II; at the beginning of the trap handler. The maximum number of software breakpoints that can be set is 256.

As described earlier, the V option in USD3.INS file, may be used to alter the default address of the trap handler. If the trap vector or handler is in the ROM location, USD-III will disable software breakpoint automatically. The user can use USD-III OPTIONS command (Sections 5.1.17/ 5.2.7) to re-enable software breakpoint after mapping trap vector and handler in RAM location.

Chapter 3

CONVENTIONS OF NOTATION AND COMMANDS SUMMARY

MICROTEK INTERNATIONAL INC.

3.1 ABOUT THIS CHAPTER

This chapter provides an overview of the conventions used with USD-III commands syntax as summarized in Section 3.2. Detailed command description and examples for each of these commands, are provided in Chapter 4.

Most of the MICE commands are transparent to USD-III and can be directly keyed-in into "COMMAND" viewport as with USD-III commands. Menu option is also provided for execution of a number of MICE commands and most of the USD-III commands through USD-III menus. The menu commands are separately discussed in Chapter 5.

3.1.1 Conventions

USD-III command syntax uses the same conventions as MICE-III commands. User should be familiar with the Conventions of Notation described in any of the MICE-III user's manuals to better understand the meaning of the command notations used in this chapter.

Frequently used key definitions are displayed in the screen layout Message Line. Refer to Chapter 1 for detailed discussions of these key definitions.

3.2 COMMAND SUMMARY

The following tables summarizes the command keywords and syntax of USD-III commands. Note that except for "DOUble", "FLoat", "HELP" and "HISTORY", all these commands may be also invoked through USD-III menus.

COMMANDS SUMMARY

3.2.1 Data Monitor Commands

| FUNCTION | FULL SYNTAX | |
|------------------------|----------------|--|
| | KEYWORD | PARAMETERS |
| Set Monitor Variable | MONitor | [{[%module]}%symbol hex_address} [number] [Byte Word Long]] |
| Clear Monitor Variable | NOMon | [no1 [no2]] |

These commands are used to set and clear monitored variables which are displayed in the DATA viewport.

Details for these commands are discussed in Sections 4.1.17 and 4.1.18. The equivalent menu commands are found under "Memory" pull-down menu in Chapter 5 or Section 5.2.3.

3.2.2 File Management Utility Commands

| FUNCTION | FULL SYNTAX | |
|----------------------|-----------------|--|
| | KEYWORD | PARAMETERS |
| Download File | Download | filename [UP UD SP SD P R] |
| Execute Command File | INClude | filename [(arg0,...,arg9)] |
| Log Command File | LOG | [filename [APPEND] CLOSE] |
| Log Output File | Journal | [filename [APPEND] CLOSE] |
| Upload File | UPLoad | filename [I [X P] T [X P] M UP UD SP SD] |

These utilities are used to download and upload files to and from MICE, as well as in creating command files and in logging all COMMAND viewport activities. Sub-commands applicable in command file are also provided in this utility and are described in detail in sub-sections under "INClude" command. These commands are described individually in Chapter 4.

Equivalent menu command for this group is found under "File" pull-down menu as explained in Section 5.2.1.

3.2.3 Floating Point Commands

| FUNCTION | FULL SYNTAX | |
|--|-----------------|--|
| | KEYWORD | PARAMETERS |
| Double Floating Point Single Floating Point | DOUble FLoat | [Physical] [address [data-1 [...data-8]]] [Physical] [address [data-1 [...data-8]]] |

USD-III uses IEEE format (Figure 3-2) to express a floating point value in binary form known as "normalized". Normalization involves adjusting the exponent in such a way that the "binary point" (analog of the decimal point) in the mantissa always lies to the right of the most significant non-zero digit. Constraining the floating point number to be in normalized form, results in a mantissa whose most significant binary digits is always 1. The IEEE format takes advantage of this by not storing this bit at all. It supports two types of precision floating point value: "DOUble" as a double-precision (64-bit) and "FLoat" as single-precision (32-bit).

This group of commands are further described in Sections 4.1.6 and 4.1.8.

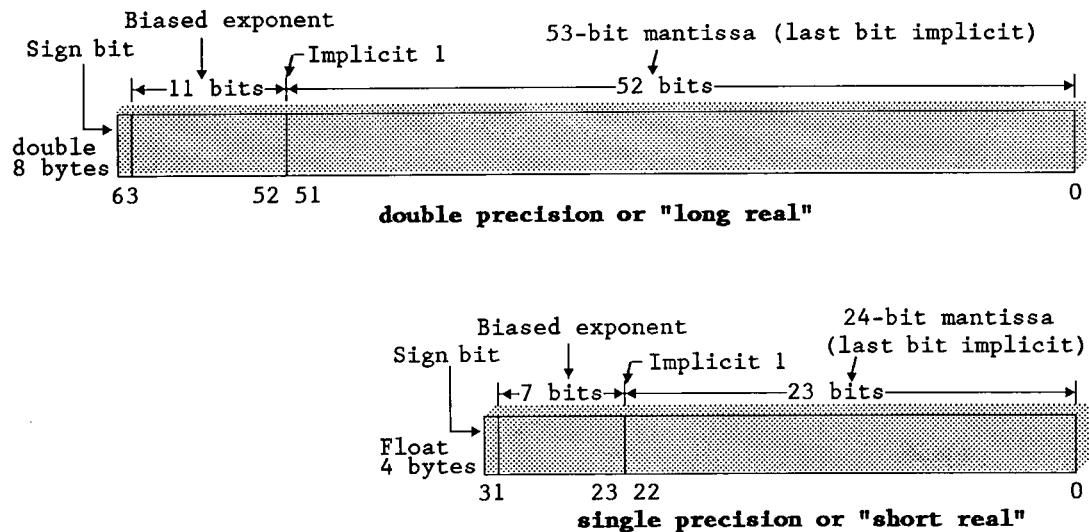


Figure 3-1 IEEE Format for Binary Representation of Floating Point Numbers

3.2.4 Software Breakpoint Commands

| FUNCTION | FULL SYNTAX | |
|----------------------|-------------|-------------|
| | KEYWORD | PARAMETERS |
| Set S/W Breakpoint | BREAKPOINT | [address] |
| Clear S/W Breakpoint | CLEAR | [no1 [no2]] |

The software breakpoint commands sets a software execution breakpoint which will halt program execution each time the target CPU attempts to execute an instruction at the specified memory location(s). When setting breakpoints, the USD-III automatically assigns each breakpoint with a reference number, i.e. "1" for the first breakpoint, "2" for the second breakpoint, so on and so forth. The reference number (not the breakpoint address) is used when clearing the breakpoint. A total of 256 software breakpoints can be set. Breakpoint settings are displayed in the BREAKPOINT viewport.

Details for these commands are discussed in Sections 4.1.1 and 4.1.2. The equivalent menu commands are found under "Trace" pull-down menu of Chapter 5 or Section 5.2.5

3.2.5 Symbol Processing Commands

| FUNCTION | FULL SYNTAX | |
|--------------------------------|-------------|----------------------------------|
| | KEYWORD | PARAMETERS |
| Create Symbol | CREATE | % symbol hex-address |
| Delete Symbol | DELETE | % symbol |
| Load Symbol Table | LOAD | filename |
| Directory Module | MODULE | |
| Symbol/Address Query | QUERY | {[% module]% symbol hex-address} |
| Include User Symbols from File | SRECALL | filename |
| Save Symbols to File | SSAVE | filename |
| List Symbol | SYMBOL | [All Global %% module] |

Symbol processing provides symbolic debugging information through the USD-III symbol tables to facilitate program development and debugging operation. It performs translation activities between symbols and address in hex values. Symbols entered as part of commands are looked up in the symbol table and the real values are passed to MICE. Address outputs from MICE are also checked against the symbol table; and when a match is found, the symbol is displayed in place of the address.

Symbols are case sensitive, i.e. upper and lower case characters are recognized as non-equivalent input and denotes separate or different symbol. Hence they are always prefixed with a percent sign <%> in order to distinguish it from the hex value in the MICE command, or from the instruction operand in the line assembler.

Symbol name should not exceed 40 characters in upper or lower case. Symbol over 40 characters are truncated. Symbol processing commands are further described individually in Chapter 4. Menu commands of this group may be found in "Symbol" pull down menu explained in Chapter 5 or Section 5.2.7.

3.2.6 Viewport Processing Command

| FUNCTION | FULL SYNTAX | |
|--------------------------------|--------------------|---|
| | KEYWORD | PARAMETERS |
| Enable/Disable Viewport Update | UPDate | [Breakpoint CODE Register Stack Trace} {ON OFF}] |

There are seven viewports in USD-III. Three of them (REGISTER, CODE and COMMAND) are displayed by default (see Figure 1-2) when USD-III is invoked. The other three (DATA, BREAKPOINT, TRACE and STACK) has to be "poped-up" on the screen. Refer to Chapter 1, Section 1.3 for function keys used in retrieving the non-default viewports. Other viewport processing keystrokes are also explained in the same section.

Because the real-time updating of viewports is a time consuming execution, "UPDate" command is provided to allow user to disable viewport update feature. This is further explained in Section 4.1.22 and in the "Options" pull-down menu in Chapter 5, Section 5.2.7.

3.2.7 Command Processing Utility

| FUNCTION | FULL SYNTAX | |
|---|--|---|
| | KEYWORD | PARAMETERS |
| USD-III Command Help Command History Instruction Step Optional Functions | HElp HIStory IStep OPTions | USD3 [number] [Over][count FOrever] [{Symbol Log Journal Breakpoint} {ON OFF}] {K1 K2 K3 K4 K5 K6 K7 K8 K9 K10} command_description [dos-command] |
| Define Softkey Temporarily exit to DOS or execute DOS command Quit USD-III Recall window/softkey Save window/softkey | SOFtkey DOS QUIT URECall USAve | [W K] [W K] |

Command Processing Utility commands are tools intended to help and provide users with the most convenient way of executing the USD-III commands. These commands are further discussed in Chapter 4. They are also available in menu commands as explained separately in the "File" menus (Section 5.2.1), "Emulation" menus (Section 5.2.4) and "Options" menu (Section 5.2.7) of Chapter 5.

Chapter 4

COMMAND DESCRIPTION AND EXAMPLES

MICROTEK INTERNATIONAL INC.

4.1 ABOUT THIS CHAPTER

This chapter describes in detail the command syntax used to operate the USD-III. The commands are sequenced in alphabetical order with one or more typical examples provided for each.

4.1.1 Set Software Breakpoint - BREAKPOINT**BREAKPOINT [address]**

BREAKPOINT is the command to set a software execution breakpoint. If "BREAKPOINT" is entered without any parameter, breakpoints arranged and consecutively numbered according to the sequence they were entered, will display.

address is the memory location of software execution breakpoint. Address may be expressed in hex value or with any USD-III symbol format, i.e. [%module%symbol].

Example: Set software breakpoints at address 2000:1234 and 2000:1240.

```
>BRE 2000:1234<CR>
>BRE 2000:1240<CR>
>BRE<CR>
S/W breakpoint toggle is ON. EV5 is reserved for USD-III.
The trap vector at location C. The handler routine at location
3FC.
1. 2000:1234      2. 2000:1240
```

The same command may be also executed from "TRACE" pull-down menu (Section 5.2.5.1.)

4.1.2 Clear Software Breakpoint - CLEar

CLEar [no1 [no2]]

CLEar is the command to delete one or more software breakpoints. If no specific breakpoint number is specified, all breakpoints are deleted.

no1 is the reference number of a particular breakpoint to be deleted or where a series of breakpoints to be deleted is to begin.

no2 is the reference number of the breakpoint where a series of breakpoints to be deleted ends.

Whenever breakpoint(s) is deleted, the breakpoint listing is automatically resorted to maintain the consecutive sequence of the breakpoint reference numbers.

Example 1: Clear the software breakpoint number "4".

```
>BRE<CR>
S/W breakpoint toggle is ON. EV5 is reserved for USD-III.
The trap vector at location C. The handler routine at location
3FC.
1. 2000:1234    2. 2000:1240
3. 3000:112    4. 3000:656
5. 3000:700
>CLE 4<CR>
>BRE<CR>
S/W breakpoint toggle is ON. EV5 is reserved for USD-III.
The trap vector at location C. The handler routine at location
3FC.
1. 2000:1234    2. 2000:1240
3. 3000:112    4. 3000:700
```

Example 2: Clear the S/W breakpoint numbered 1, 2, 3

```
>CLE 1 3<CR>
>BRE<CR>
S/W breakpoint toggle is ON. EV5 is reserved for USD-III.
The trap vector at location C. The handler routine at location
3FC.
1. 3000:700
>
```

The same command may be also executed from "TRACE" pull-down menu (Section 5.2.5.1.)

4.1.3 Create Symbol - CCreate

```
CCreate %symbol hex_address
```

CCreate is the command to create a global symbol.

symbol is the symbol to be created with any of the alphanumeric characters except the percent sign <%>.

hex_address is the symbol address. If the segment value is omitted from the command line and it is a segmented address format, the default CS value will be used as the segment value.

Any symbol created by this command is temporary and effective only during the current session. It will not be written back to the symbol file but may be saved as a disk file using the SSave command explained in Section 4.1.24. Note that the maximum number of symbols that can be save in a symbol table is 300.

Example: Create symbols "SYM1 12:34" and "SYM2 FF".

```
>CCreate %SYM1 12:34<CR>
Symbol SYM1 [0012:0034] is created.
>CCreate %SYM2 FF<CR>
Symbol SYM2 [F000:00FF] is created.
>
```

The same command may be also executed with the "Create" menu command from "SYMBOL" pull-down menu (Section 5.2.6.2.)

4.1.4 Delete Symbol - DELetE**DELetE %symbol**

DELetE is the command to delete an existing symbol. Only symbols created by "CReate" command can be deleted with this command.

symbol specifies the symbol to be deleted.

Example: Delete symbols "SYM1" and "SYM2".

```
>DEL %SYM1 [0012:0034]<CR>
    Symbol SYM1 [0012:0034] is deleted.
>DEL %SYM2 [F000:00FF]<CR>
    Symbol SYM2 [F000:00FF] is deleted.
>
```

The same command may be also executed from "SYMBOL" pull-down menu (Section 5.2.6.3.)

4.1.5 Temporary Exit to DOS - DOS

DOS [dos-command]

DOS allows temporary exit to DOS without quitting USD-III.

The same command may be also executed from "FILE" pull-down menu (Section 5.2.1.)

4.1.6 Double Floating Point - DOUble

```
DOUble [Physical] [address [data-1 [...data-8]]]
```

DOUble is the command for Memory Modify and specifies the desired data format as double-precision (64-bit).

Physical classifies type of address as "physical" for MICE-16 80286 and MICE-32 80386. Otherwise, the address is "logical".

address specifies the location where the Memory Modify operation is to begin. It is a hexadecimal address in the memory space of the emulation CPU.

Address may be expressed in hex value or in any USD-III symbol format, i.e. [%module]%symbol.

data-1...8 are the floating-point values. Up to 8 values may be entered. A floating-point value consists of the following:

[sign] integer_part [.fractional_part] [exponent]

where:

sign is an optional plus (+) or minus (-) sign. Note that when no sign is entered, it is understood to be plus (+).

integer_part is a single or a series of decimal digits.

.fractional_part is an optional decimal fraction consisting of a zero or more decimal digits following the dot (.).

exponent is an optional "e" or "E" exponent followed by an integer value prefixed with optional plus (+) or minus (-).

e.g., 76.3e-14; 76.3E+5; -76.3E2; 76.e-8; 76.3; 76.0; 76.; 76

If no parameter is entered, the USD-III displays the starting address content of the current PC and waits for further input. To change the value held in the location displayed, enter a new floating-point value, then press <CR> to go to the next location in memory or <Ctrl-J> to go to the preceding location. Pressing <CR> or <Ctrl-J> alone, without first inputting a new value, leaves the contents of the location unchanged; pressing <Esc> terminates the function.

Example: **>DOU<CR>**

| | | |
|---|--------------------|--------------------------|
| 0000:0002 | -6.8285270344e-229 | <u>2<CR></u> |
| 0000:000A | -6.595736438e-229 | <u>3.4<CR></u> |
| 0000:0012 | -6.8285270344e-229 | <u>6E10<CR></u> |
| 0000:001A | -6.8285270344e-229 | <u>11<CR></u> |
| 0000:0022 | -6.8285270344e-229 | <u>123.456<CR></u> |
| 0000:002A | -6.8285270344e-229 | <u><LF></u> |
| 0000:0022 | 123.456 | <u><LF></u> |
| 0000:001A | 11. | <u><LF></u> |
| 0000:0012 | 6.e+010 | <u><LF></u> |
| 0000:000A | 3.4 | <u><LF></u> |
| 0000:0002 | 2. | <u><LF></u> |
| F000:FFFFA | -7.0079635217e-229 | <u><LF></u> |
| F000:FFF2 | -3.5408733685e+279 | <u><ESC></u> |
| >DOU 10 1.2 2.3 3.4 4.5 5.6<CR> | | |
| > | | |

4.1.7 Download File - Download**D**ownload **filename [UP|UD|SP|SD|P|R]**

Download is the command to download an object file to MICE.

filename is the file to be downloaded

UP|UD|SP|SD are the four optional memory space types of MICE-16 68000:

- User Program (UP)
- User Data (UD)
- Supervisor Program (SP)
- Supervisor Data (SD)

Hence, these options are applicable and to be specified only when USD-III is linked with MICE-16 68000. If no parameter is entered, "SP" will default.

P|R is the instruction to load MICE-32 80386/MICE-16 80286 Descriptor Table and enter into "P" (Protected) Mode, or to download data in a real (R) mode. Hence these parameters are applicable to MICE-32 80386/ MICE-16 80286 only.

To download an object file to MICE memory, input Download followed by the name of the file to be downloaded. The file may be of an Intel hex, Intel binary (OMF), Motorola S or Tektronix hex. After the object file is downloaded, the symbol table files specified by the download filename, are then automatically loaded immediately if exist.

To abort downloading after it has been initiated, press <Esc> and the message-

"**DOWNLOAD ABORTED!**"

will display.

Also note that if downloading is unsuccessful, an error message-

"**DOWNLOAD FAILURE!**"

will display and USD-III will exit from this command.

Example 1: Download file "T80186.HEX".

```
>Download T80186.HEX<CR>
Download in process:45F3
Download completed!
>
```

Example 2: Download file "MRI680CX.ABX" into MICE-16 68000 accessing user program (UP) memory space type.

```
>Download MRI680.ABX UP<CR>
Download in process : 001EB8
Download completed!
>
```

The same command may be also executed from "FILE" pull-down menu (Section 5.2.1.1.)

4.1.8 Single Floating Point - FLoat

FLoat [Physical] [address [data-1 [...data-8]]]

FLoat is the command for Memory Modify and specifies the desired data format as single-precision (32-bit).

Physical classifies type of address as "physical" for MICE-16 80286 and MICE-32 80386. Otherwise, the address is "logical".

address specifies the location where the Memory Modify operation is to begin. It is a hexadecimal address in the memory space of the emulation CPU.

Address may be expressed in hex value or in any USD-III symbol format, i.e. [%module]%symbol.

data 1...8 are the floating point values described in the preceding section.

If no parameter is entered, the USD-III displays the starting address content of the current PC and waits for further input. To change the value held in the location displayed, enter a new floating point value, then press <CR> to go to the next location in memory or <Ctrl-J> to go to the preceding location. Pressing <CR> or <Ctrl-J> alone, without first inputting a new value, leaves the contents of the location unchanged; pressing <Esc> terminates the function.

Example: >FL 0:200<CR>

| | | |
|---|--------------------|-------------|
| 0000:0200 | -1.7013336459e+038 | <CR> |
| 0000:0204 | -6.5415813844e+035 | <CR> |
| 0000:0208 | -2.5145813338e+033 | 12<CR> |
| 0000:020C | -9.6635056222e+030 | 34<CR> |
| 0000:0210 | -3.7126671521e+028 | 56.1234<CR> |
| 0000:0214 | -1.4259872736e+026 | <CR> |
| 0000:0218 | -5.4754450819e+023 | <Esc> |
| > <u>FL 0:200 1.1 2.2 3.4<CR></u> | | |
| > | | |

4.1.9 USD-III Command Help - HEIp**HEIp USD3**

HEIp is the command to display full screen "Help" display which will include USD-III command syntax and function-key definitions under default screen.

USD3 is the required parameter to displays the full screen "Help" display.

Pressing <F1> while in default window status, will also show the same full screen "Help" display.

4.1.10 Command History - HIStory

HIStory [number]

HIStory is the command to execute a specific command stored in history buffer. "HIStory" alone without parameter will display the stored commands arranged and numbered in chronological order.

number is the reference number (1 to 32) of the specific command stored in the history buffer.

USD-III provides a history buffer where up to 32 of the most recently executed commands are stored. Commands of the history buffer may be selected to be executed. This eliminates the need to constantly re-key the frequently used commands.

Example: Display content of Command History buffer and repeat line command number 3.

```
>HIS<CR>
1 DOWNLOAD DEMO.S37
2 MODULE
3 M 0 FF
>HIS 3<CR>
>Memory 0 FF
:
:
```

4.1.11 Execute Command File - INClude

INClude filename [(arg0,...,arg9)]

INClude is the command to execute a user-defined command file.

filename is the command file to be executed.

arg0,...arg9 are the arguments (MICE/ USD-III commands, command parameters, sub-commands, nested commands, macros, values, etc.) defined to substitute the corresponding variables (%0, %1, %2,...%9) specified in the command file.

When "INClude" command is executed, the variables (%0,..%9) specified in the command file are substituted with the argument in the same sequence they appeared in the command line as shown below. Note that arguments are to be entered separated by commas and enclosed in parenthesis-

>INC INITIAL.INC(100,10,200,8F,0,3FFFF)

When executed, "100" replaces "%0", "10" replaces "%1", "200" replaces "%2", etc., in the command file.

NOTE

Nested command files (see Section 4.1.11.9) can be specified but is limited to a maximum of one level nesting only, e.g. the nested command file should NOT contain any other "INClude" sub-command.

The following is summary of the USD-III pre-defined sub-commands which are utilized in creating command files. Details of this sub-commands are further explained in the sub-sections following this section.

| | |
|-------------------------------|--|
| \ | - used to substitute <Esc> in command file. |
| %0 - %9 | - the replaceable variables when executing the command file. |
| BELL [count] | - rings the bell within executed command file. Bells can be timed by [count] parameter. |
| ECHO [ON OFF] | - toggles screen display of command in the command file. |
| ERROR {ABORT CONTINUE} | - determines the action USD-III should take when an error occurs in executing command file. |
| GET str | - instructs USD-III to receive keyboard input |
| GOTO label | - transfers program control to the line following the specified label. |
| IF[_NOT] / ELSE | - permits conditional execution of commands in the command line. |
| INClude | - Nest command file. |
| LOOP | - specifies the beginning of a loop block |
| LOOPEND [count] | - specifies the end of loop block, and the loop count. |
| PAUSE [str count] | - specifies a checkpoint where execution pauses, and defines a remark string or pause time. |
| REMark str | - specifies a comment string. |
| TICK [count] | - determines the time interval between the MICE-free-run command and its subsequent command. The interval is in number of seconds. |
| :label | - specifies a label to be referred to by "GOTO" sub-command. |

The same command may be also executed from "FILE" pull-down menu (Section 5.2.1.3.)

4.1.11.1 Substitute for <Esc> in Command File - <\>

```
\
```

<\> is the character to use in place of <Esc> in the Command File.

Example: Define Step command and terminate after six lines of data are displayed.

```
S<CR>
<CR>
<CR>
<CR>
<CR>
<CR>
\
```

4.1.11.2 Variables to be Replaced - %0 - %9**%0 - %9**

%0 - %9 are the variables inserted in the command file which are to be replaced by the arguments specified in the INClude command line (Section 4.1.11).

Example: Assign variable %0 to %4 in the command file "DEMO.SUB",

```
•  
•  
•  
RES %0  
FILL %1 %2 %3  
ISTEP %4  
R  
•  
•
```

then execute the command file with arguments (FFF0,0,F,90,3)-

>INC DEMO.SUB(FFF0,0,F,90,3)<CR>

the command file actual execution will be-

```
•  
•  
•  
RES FFF0  
FILL 0 F 90  
ISTEP 3  
R  
•  
•
```

4.1.11.3 Ring the Bell - BELL

BELL [count]

BELL is the sub-command to activate the computer's bell (sounds a short beep).

count is the number of times the computer has to beep.

This command may be inserted anywhere within a command file to ring the computer's bell (speaker). This function may be used as audio indicator to notify the user that a specified procedure has been completed, to call user's attention to an error or start observing a marked key points of a data, to remind user that further input is required to continue execution, etc.

If no parameter is entered, one beep will default.

Example: Beep the bell 3 times when "ERROR" occurs.

```
:
:
:ERROR
BELL 3
:
:
```

4.1.11.4 Command Display Control - ECHO**ECHO [ON|OFF]**

ECHO is the sub-command to toggle screen display of commands in the command file ON and OFF

ON|OFF are the options to turn display ON or OFF respectively.

Normally, commands in the command file are displayed (echoed) on the screen as they are executed by USD-III. The display may be turned off by using "OFF" option following ECHO command. Similarly, the display may be turned ON with the "ON" option. If no option is entered, previous setting will default. Otherwise USD-III default, "ECHO On" will prevail.

Example: Turn ECHO OFF and then ON again in command file.

```
ECHO OFF
RES 0
FILL 0:0 FF 90
J 0:0
ECHO ON
S
<CR>
<CR>
\
```

4.1.11.5 Abort/ Continue Execution in Case of Error - ERROR**ERROR {ABORT|CONTINUE}**

ERROR is the sub-command to stop or let USD-III continue executing the next command in the executed command file whenever error is detected during such execution.

ABORT|CONTINUE is the options to stop or continue execution of the error confronted command file.

The **ERROR** command is inserted in the command file to determines which action USD-III should take whenever error is detected while executing a command file. If "ABORT" is specified, USD-III will terminate the execution of the command file. When "CONTINUE" is used, USD-III will ignore such error and continue to execute the command file. If **ERROR** sub-command is not inserted, "ERROR CONTINUE", will default.

Example: Define execution to continue if error occurs before ":STEP 2" and stops if error occurs after ":STEP 2"

```
ERROR CONTINUE
J 0
S
<CR>
<CR>
\
G
IF NOT "NOT READY" RES
:STEP_2
ERROR ABORT
C
<CR>
<CR>
<CR>
\
```

4.1.11.6 Prompt for Keyboard Input - GET**GET str**

GET is the sub-command to instruct USD-III to pause execution of command file and prompt user for a keyboard input.

str is a string of user specified message or comment to remind and prompt user to enter a predefined input.

This sub-command is most useful in providing user with more options for specifying conditional flow in addition to "IF" and "ELSE".

Example: Define the following "GET" sub-command line in the command file.

```
GET Please key in any character:  
IF GET=1 goto sub1  
IF GET=2 goto sub2  
IF GET=3 goto sub3
```

When executed, the result will be:

```
:  
:  
Please key in any character: 1<CR>  
Jump 200 ;The command is now under label sub1  
:  
:
```

4.1.11.7 Redirect Flow of Command Execution - GOTO

GOTO label

GOTO is the sub-command to redirect the flow of a command execution to jump forward or backward in order to accomplish a specified task instead of continuing to the next command line.

label is the reference or "labeled" point where the command execution flow is redirected to. Labels may consist of up to 10 characters, and must be prefixed by a colon [:].

Note that labels are never displayed during program execution. Non-referenced labels may therefore be used to place comments within the command file which do not display during execution. Total labels within a command file should not exceed 100 including nested command files.

Example: Redirect flow of command execution to "NEXT".

```
:  
:  
GOTO NEXT  
:  
:  
:  
:  
:NEXT
```

4.1.11.8 Conditional Execution - IF / ELSE

IF[_NOT] condition command
ELSE condition command

IF / ELSE is the sub-command that permits conditional execution of a command line.

_NOT is the optional parameter that will limit command execution to "condition" which is NOT true.

condition is a string enclosed in quotation marks ["]. If the string appears in the last command output, the resulting condition is TRUE. Otherwise the resulting condition is FALSE.

command is a MICE or USD-III command to be executed when the statement is matched. Otherwise, execution skips to the next command line.

Example: Define a negative condition to check Register D0.

```
R DO
 \
 IF NOT "D0=00000030" GOTO SHORT
```

Note that the blank spaces between two elements of a conditional string may be replaced by the equal sign [=], as shown in the above example where exact string syntax for register display is "D0 00000030".

4.1.11.9 Nest Command File - INClude**INClude filename [(arg0,...,arg9)]**

INClude is the sub-command to nest a command file within another command file..

filename is the command file filename to be nested. This filename must be different from "parent" command file.

arg0,...,arg9 are the arguments defined to substitute corresponding variables (%0,...%9) as described in Section 4.1.11.

Note that nested command files is limited to one level nesting only, e.g. the nested command file should NOT contain any other "INClude" sub-command.

Example: Specify "SAM68" as nested command file after Register command.

```
M 0 FF
R
INC SAM68
:
:
```

INClude - :label

COMMAND EXAMPLES

4.1.11.10 Label to be Referred to by "GOTO" Sub-Command - :label**:label**

:label Specifies a label to be referred to by "GOTO" sub-command as explained in Section 4.1.11.7.

Example: see Section 4.1.11.7.

4.1.11.11 Loop Block - LOOP / LOOPEND

| |
|--|
| Beginning of a Loop Block - LOOP |
| End of a Loop Block - LOOPEND [count] |

LOOP is the sub-command for specifying the beginning of a loop block.

LOOPEND is the sub-command for specifying the end of a loop block.

count is the hexadecimal value indicating the number of loops (maximum value is 0FFH) desired. If count is not specified, the default is 1.

Example: Define a loop block to check Register D0 and D1 values during Stepping and loop the block 8 times.

```
LOOP
R D0
\
IF NOT "D0=00000030" GOTO SHORT
R D1
\
IF "D1=00000041" GOTO MATCH
S

\
LOOPEND 8
```

4.1.11.12 Execution Checkpoint - PAUSE**PAUSE [str|count]**

PAUSE specifies a checkpoint where a command file execution pauses to allow user to observe the screen display. Note that checkpoints must be defined as a separate command and should not be entered within any other command. If only PAUSE is entered without any parameter, USD-III will display a message (see below) at the specified checkpoint.

str is a user-defined character string which will display when execution reaches the specified checkpoint.

count is the user assigned pause time (in seconds). When the time elapsed, command execution resumes automatically.

When [string] parameter is opted, execution of the command file will pause at the indicated checkpoint and display the user defined string followed by the following message:

"PRESS <ESC> TO TERMINATE COMMAND FILE, ANY OTHER KEY TO CONTINUE"

The same message will display if only PAUSE is entered without any parameter.

Checkpoints are usually defined at appropriate locations in the file where disks need to be changed between commands.

Example: Define a PAUSE command with "TEST OK!" as string.

```
:START
R AO
\
IF NOT "AO=00000000" GOTO ERROR
PAUSE TEST OK!
GOTO START
:ERROR
BELL 5
```

4.1.11.13 Comment String - REMark**REMark str**

REMark is the sub-command to display a user defined remark.

str is the user-defined string of remark to be displayed.

Example: Define a remark "*condition not matched!*" in the command file.

```
:  
:  
REM condition not matched!  
:  
:
```

**4.1.11.14 Set Time Delay Between MICE
Free-Run and its Subsequent Command - TICK****TICK [count]**

TICK is the sub-command to set an execution time delay between a MICE free-run command and its subsequent command.

count is a user defined hexadecimal value, 1H ~ 7FFFH (1 ~ 32767) specifying the time interval (in seconds) between commands.

TICK sub-command is applicable for setting time delay between MICE free-run and its subsequent command. It is most useful particularly when user sets and failed to hit a breakpoint while target is free running. Without the time delay, execution of a free-run command with "a distant address breakpoint", may prematurely ends at the next (Register, Memory, etc.) command before it hits the breakpoint. With the presence of TICK sub-command, the command file will automatically delays fetching of the next command line until the set time elapses. As it time-outs, the MICE prompt appears, and execution continued to the next command line. The further the breakpoint setting is, the longer the time delay should be.

Example: Set a TICK sub-command with 8 seconds delay between 'GO' command and its subsequent (Register) command in the command file.

```
TICK 8
Go
R
```

4.1.12 Instruction Step - IStep**IStep [Over] [count|FOrever]**

IStep is the command for instruction step. This command executes instruction by single-stepping and returns to command prompt mode after executing the command. After each step, After each step, the viewports information are also updated accordingly if their respective updating toggle is enabled.

Over is the option to treat subroutines as normal instructions. With this option, user needs not step into subroutines which he is not concerned about. USD-III will run through such subroutines.

NOTE

Note that this option is NOT applicable to MICE-II.

count is a hexadecimal value from 1H to 7FFFFFFH specifying the number of times the instruction program has to step.

FOrever is the option to step instruction continuously until forced to stop by pressing <Esc> key.

User may increase stepping speed by disabling viewport update feature on some viewports as described in Section 4.1.22.

Example: Execute 3 steps

```
>IStep 3<CR>
3 step!
>
```

The same command may be also executed from "EMULATION" pull-down menu (Section 5.2.4.)

4.1.13 Log Output File - JOurnal**JOurnal [filename [APPEND]|CLOSE]**

JOurnal is the command to log all of the COMMAND viewport activities (input and response) to an assigned filename as they took place during a debugging session. "JOurnal" alone without any parameter will display the journal filename and the journal toggle status.

filename is the file created to store the logged COMMAND viewport activities.

APPEND is option to append the newly logged activities into an existing file. Otherwise existing data in the file will be automatically overwritten.

CLOSE is the option to save the logged COMMAND viewport activities into the assigned filename, and then end the logging session.

Example: Log the following commands and their responses to "DEMO" file.

```
>JO DEMO<CR>
>D 0 F<CR>
:
:
>R <CR>
:
:
>JO CLOSE<CR>
```

the "DEMO" file content will be :

```
>D 0 F
:
:
>R
:
:
```

The same command may be also executed from "FILE" pull-down menu (Section 5.2.1.5.)

4.1.14 Load Symbol Table - LOAd**LOAd filename**

LOAd is the command to load symbol file from disk into USD-III.

filename specifies the symbol file to be loaded.

Example: Load Symbol file "TEST".

```
>LOA TEST<CR>
  Symbol table is loaded OK from file TEST.TBL and TEST.SYB.
>
```

The same command may be also executed from "SYMBOL" pull-down menu (Section 5.2.6.1.)

4.1.15 Log Command File - LOG**LOG [filename [APPEND]|CLOSE]**

LOG is the command to log all of the commands to an assigned filename as they are entered and executed during a debugging session. "LOG" alone without any parameter will display the log filename and the log toggle status.

filename is the file created to store the logged commands.

APPEND is the option to append the newly logged commands into an existing file. Otherwise existing data in the file will be automatically overwritten.

CLOSE is the option to save the logged commands into the assigned file, and then end the logging session.

Example: Log the following command into "DEMO" file.

```
>LOG DEMO<CR>
>D O F<CR>
:
:
>R <CR>
:
:
>LOG CLOSE<CR>
```

the "DEMO" file content will be:

D O F
R

The same command may be also executed from "FILE" pull-down menu (Section 5.2.1.4.)

4.1.16 Directory Module - MODULE**MODULE**

MODULE is the command to display all modules existing in the symbol file.

Example: Display all existing modules.

```
>MODU<CR>
  1. TEST03
  2. PLMTEST
  3. TEST032
>      The total number of modules: 3.
```

The same command may be also executed from "SYMBOL" pull-down menu (Section 5.2.6.)

4.1.17 Set Monitor Variable - MONitor

| |
|---|
| MONitor {[%module]%symbol hex_address} [number] [Byte Word Long] |
|---|

MONitor is the command to set variables to be displayed and monitored on the DATA viewport.

[%module]%symbol is the corresponding symbol name of the address to be monitored. Module name in "[%module]%symbol" symbol format may also be specified, but only the specified module will be searched.

hex_address is the hexadecimal address to be monitored.

number is the number of elements to be monitored on the DATA viewport. These elements are to begin at the specified address (indicated as +1, +2, etc., in the DATA viewport). The unit of each element is defined in the next (either Byte, Word or Long) argument.

Byte|Word|Long is the type of monitored variable. The default is Byte.

Example:

1. Display 10 bytes of data beginning at address "0EF50".

>MONitor 0EF50 10<CR>

2. Display 5 words of data beginning with symbol "array_1".

>MONitor %array_1 5 w<CR>

The command may be also executed with "Data Monitor" command in the "MEMORY" pull-down menu (Section 5.2.3.)

4.1.18 Clear Monitor Variable - NOMon**NOMon [no1 [no2]]**

NOMon is the command to delete one or more monitored variables from DATA viewport display. If no specific variable is specified, all monitored variables are deleted.

no1 is the reference number of a particular monitored variable to be deleted or where a series of to be deleted monitored variables, begins.

no2 is the reference number where a series of to be deleted monitored variables, ends.

Whenever any of the numbered variables are deleted, the remaining variables on display are automatically resorted to maintain the consecutive sequence of the variables.

Example: 1. Clear the variable number 4.

>NOMon 4<CR>

2. Clear the variables number 3 to 5.

>NOMon 3 5<CR>

The command may be also executed with "Data Monitor" command in the "MEMORY" pull-down menu (Section 5.2.3.)

4.1.19 Optional Features - OPTions**OPTions [{Symbol|Log|Journal|Breakpoint} {ON|OFF}]**

- OPTions** is the command to enable/ disable available USD-III optional features. If no parameter is entered, current "OPTions" status will display.
- Symbol** is the option which provides an ON/ OFF toggle to enable/ disable the Symbolic Debugging feature.
- Log** is the option which provides an ON/ OFF toggle to enable/ disable the Log feature.
- Journal** is the option which provides an ON/ OFF toggle to enable/ disable the Journal feature.
- Breakpoint** is the option which provides an ON/ OFF toggle to enable/ disable the S/ W Breakpoint feature.
- ON|OFF** is the toggle to enable or disable the Symbol/ Log/ Journal/ S/ W Breakpoint feature.

Example: Display "OPTION" status and disable Symbol, Log and Journal features.

```
>OPT S OF<CR>
>OPT L OF<CR>
>OPT J OF<CR>
>OPT <CR>
SYMBOL toggle: OFF          LOG toggle :      OFF
JOURNAL toggle: OFF         S/W BREAKPOINT toggle: ON
>
```

The same command may be also executed from "OPTIONS" pull-down menu (Section 5.2.7.)

4.1.20 Symbol/Address Query - QUEry**QUEry {[%%module]%symbol|hex_address}**

QUEry is the command to query for the address of a specified symbol or the symbol name of a specified address. USD-III will search through the symbol table and list all occurrences of the specified symbol or address.

[%%module]%symbol

is the corresponding symbol name of the address in query. Module name in "[%%module]%symbol" symbol format may also be specified, but only the specified module will be searched.

hex_address is the corresponding hexadecimal address setting of the symbol name in query.

Example: Query address of symbol "SYM3" and the symbolic equivalence for address "C000:5678".

```
>QUE %SYM3<CR>
  Global newly created      Address: 0000:4321
>QUE C000:5678<CR>
  Global newly created: SYM4
>
```

The same command may be also executed from "OPTIONS" pull-down menu (Section 5.2.6.4/5.)

4.1.21 Quit USD-III - QUIT

QUIT

QUIT is the command to quit USD-III and exit to DOS.

After executing the command, a dialog-box (Figure 4-1) will display to confirm whether user wishes to **"Save screen and softkey?"**. If the answer is "Y", USD-III will execute the "USAve" command (Sections 4.1.25/ 5.2.7.4) internally before exiting. If the answer is "N", the system will immediately exit to MS-DOS.

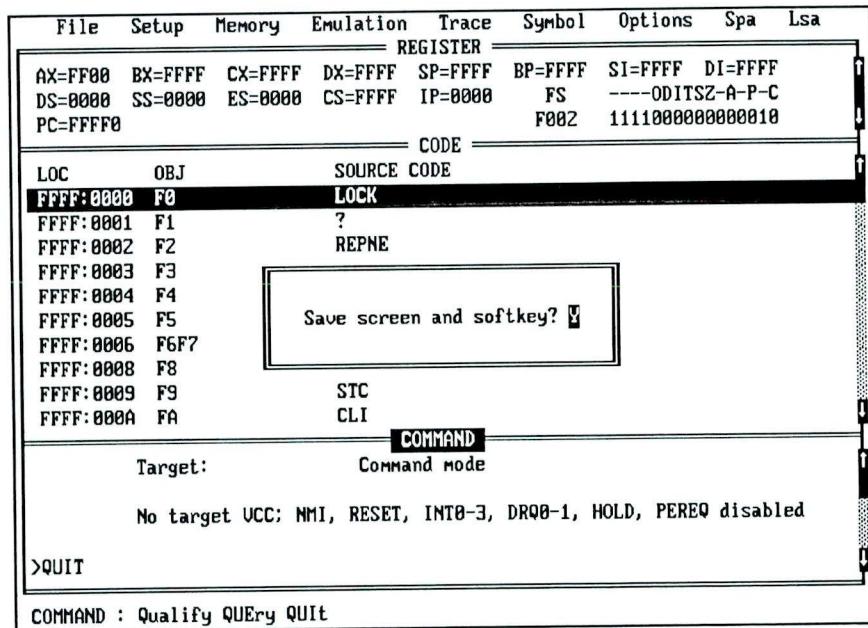


Figure 4-1 "Quit USD-III" Dialog Box

The same command may be also executed from "FILE" pull-down menu (Section 5.2.1.)

4.1.22 Define SoftKey - SOFkey

| |
|--|
| SOFkey {K1 K2 K3 K4 K5 K6 K7 K8 K9 K10} command_description |
|--|

SOFkey is the command to define a function-key operated command (softkey) to replace one or more standard commands.

K1...K10 is the name to represent the <Shift-F1> to <Shift-F10>.

command_description

specifies the command(s) to be substituted by the <Shift-F1> to <Shift-F10>. If more than one line command are specified, they should be separated with a semi-colon (;).

Example: Define softkey 'K1' for 3 commands : 1) JOURNAL TEST.DAT; 2) M 10000 100A0; 3) JOURNAL CLOSE.

```
>SOF K1 JOURNAL TEST.DAT; M 10000 100A0; JOURNAL CLOSE<CR>
>SOF<CR>
  K1 = JOURNAL TEST.DAT; M 10000 100A0; JOURNAL CLOSE
  K2 =
  K3 =
  K4 =
  K5 =
  K6 =
  K7 =
  K8 =
  K9 =
  K10=
>
```

SOFtkey

COMMAND EXAMPLES

```
><Shift-F1>
>JOURNAL TEST.DAT
>Memory 10000 100AO
    00  02  04  06  08  0A  0C  0E      ASCII-CODE
010000 4E71 4E71 4E71 4E71 4E71 4E71 4E71 NqNqNqNqNqNqNqNqNq
010010 4E71 4E71 4E71 4E71 4E71 4E71 4E71 NqNqNqNqNqNqNqNqNq
010020 4E71 4E71 4E71 4E71 4E71 4E71 4E71 NqNqNqNqNqNqNqNq
010030 4E71 4E71 4E71 4E71 4E71 4E71 4E71 NqNqNqNqNqNqNqNq
010040 4E71 4E71 4E71 4E71 4E71 4E71 4E71 NqNqNqNqNqNqNqNq
010050 4E71 4E71 4E71 4E71 4E71 4E71 4E71 NqNqNqNqNqNqNqNq
010060 4E71 4E71 4E71 4E71 4E71 4E71 4E71 NqNqNqNqNqNqNqNq
010070 4E71 4E71 4E71 4E71 4E71 4E71 4E71 NqNqNqNqNqNqNqNq
010080 4E71 4E71 4E71 4E71 4E71 4E71 4E71 NqNqNqNqNqNqNqNq
010090 4E71 4E71 4E71 4E71 4E71 4E71 4E71 NqNqNqNqNqNqNqNq
0100AO 4E71                                     Nq
>JOURNAL CLOSE
>
```

The same command may be also executed from "OPTIONS" pull-down menu (Section 5.2.7.3.)

4.1.23 Include User Symbols from File - SRecall**SRecall filename**

SRecall is the command to retrieve saved symbols from files previously created and saved with the SSave command (Section 4.1.24).

filename is the file from which the saved symbols are retrieved.

This command is useful when retrieving symbols to insert them into a newly created symbol table. If any symbols already exist in the new symbol table when the saved symbols are retrieved, the symbols from disk file are appended into the newly created symbol table. However, NOT all saved symbols may be retrieved into the new symbol table if the following conditions exists:

1. Identical symbol names exist.
Saved symbol names which already exist in the new symbol table can not be retrieved.
2. The total number of symbols in the new symbol table exceeds 300.
The number of symbols that can be retrieved into the new symbol table is limited to its available symbol slots after new symbols has been created. Saved symbols are retrieved according to their filing sequence until the total 300 symbol limit is reached. The left-over are ignored.

Example: Retrieve all the symbols saved in the file "main.sym".

>SRecall main.sym<CR>

The command may be also executed with "Srecall" menu command from "SYMBOL" pull-down menu (Section 5.2.5.)

4.1.24 Save Symbols to File - SSave**SSave filename**

SSave is the command to save the symbols created by CReate command (Section 4.1.3) to a disk file.

filename defines the filename to save the symbols. If such filename already exist, it will be overwritten. Otherwise it will be created.

A maximum of 300 symbols can be saved from a created symbol table. The save symbols may be retrieved with SRecall command explained in Section 4.1.23.

Example: Save current symbols into the file "main.sym".

>SSave main.sym<CR>

The command may be also executed with "Ssave" menu command from "SYMBOL" pull-down menu (Section 5.2.5.)

4.1.25 List Symbol - SYMbol**SYMbol [All|Global|%%module]**

SYMbol is the command to display symbols. If no parameter is given, all symbols will be listed.

All is the option to display all symbols including global symbols (from symbol file or created by "CReate" command) and symbols in each modules.

Global is the option to display global symbols from symbol file or created by "CReate command".

%%module specifies the module name under which the symbols are to be displayed.

Example: Display symbols under module "PLMTEST".

```
>SYM %%PLMTEST<CR>
Module PLMTEST : 5 symbols
0053:A3D6 B01          0053:A3D7 BA01
0053:95A2 C01          0053:95A4 CA01
0053:9344 D01
>
```

The command may be also executed with "List" command in the "SYMBOL" pull-down menu (Section 5.2.5.)

4.1.26 Enable/ Disable Viewport Update - UPDate

| |
|--|
| UPDate {Breakpoint CODE Register Stack Trace Data} {ON OFF} |
|--|

UPDate is the command to enable or disable data updates on BREAKPOINT, CODE, REGISTER, STACK, TRACE and DATA viewports.

Breakpoint|CODE|...|Data

are the viewports in which data updating feature may be enabled or disabled.

ON|OFF is the enable/ disable toggle.

NOTE

Note that disabling viewport update feature will increase USD-III execution speed.

WARNING

If user wishes to switch into protected mode (available with MICE-II 80286, MICE-16 80286 and MICE-32 80386), viewport update feature must be disabled. Otherwise unexpected errors will result. For detailed description, please refer to the pertinent MICE manual.

Example: Disabling the update feature of REGISTER viewport

>UPD R OF<CR>
>

The same command may be also executed from "OPTIONS" pull-down menu (Section 5.2.7.2.)

4.1.27 Upload File - UUpload**UUpload filename [I |X|P]|T |X|P|M|UP|UD|SP|SD]**

UUpload is the command to upload object code from MICE to a specified filename.

filename is the file to be uploaded.

I Intel hex or extended Intel hex.

T Tektronix hex.

X|P are the two types of optional memory spaces of MICE-8 8052:

- Data Memory (X)
- Program Memory (P)

Hence, these options are applicable and to be specified only when USD-III is linked with MICE-8 8052. If no parameter is entered, "P" will default.

M Motorola hex.

UP|SD|UD|SP are the four types of optional memory spaces of MICE-16 68000:

- User Program (UP)
- User Data (UD)
- Supervisor Program (SP)
- Supervisor Data (SD)]

Hence, these options are applicable and to be specified only when USD-III is linked with MICE-16 68000. If no parameter is entered, "SP" will default.

To upload object code to the specified filename, input "**UUpload**" followed by the name of the file to be uploaded and the desired memory space type. If the MICE model running on USD-III is of Intel 8086/88, 80186/188, 80286 or 80386, then the Code Segment value for the object code to be uploaded will display (see Example 2 below). Then the memory range request message-

"Upload memory range:"

will display. Respond with the range to be uploaded or press <Esc> to terminate upload command. More than one segment may be uploaded. The uploaded memory contents are then sent through MICE to a disk file.

UPload

COMMAND EXAMPLE

If <Esc> is entered or uploading is unsuccessful, a warning message-

"UPLOAD ABORTED!"

will display and the command is aborted.

Example 1: Upload file TEST.S37 from MICE-32 68020.

```
>UPload TEST.S37<CR>
    Upload memory range: 0 FF<CR>
    Upload in process:EO
    Upload memory range: 100 1FF<CR>
    Upload in process:1EO
    Upload completed!
>
```

Example 2: Upload file TEST.HEX from MICE-16 8086F.

```
>UPload TEST.HEX<CR>
    CS= FFFF
    Upload memory range: 0:100 4FF<CR>
    Upload in process:0000:04EO
    Upload memory range: 100:0 FF<CR>
    Upload in process:0100:00EO
    Upload completed!
>
```

NOTE

When uploading a file within command file, if the specified filename already exists in memory, it will be overwritten automatically; i.e. no warning message will be displayed.

The same command may be also executed from "FILE" pull-down menu (Section 5.2.1.2.)

4.1.28 Recall Window Status and Softkey Definitions - URECall**URECall [W|K]**

URECall is the command to recall window status and softkey definitions saved in the USD3.SAV file.

W|K "W" is specified if only window status, including the dialog box color setting, is to be recalled. Likewise, "K" is specified when recalling softkey definitions alone. If none is specified, both window status and softkey definitions will be recalled.

This command is useful when user wishes to recall a saved window status and softkey definitions.

The same command may be also executed from "OPTIONS" pull-down menu (Section 5.2.7.5.)

4.1.29 Save Window Status and Softkey Definitions - USAve**USAve [W|K]**

USAve is the command to save current window status (including screen layout and viewport update flag, but NOT its contents) and softkey definitions into the USD3.SAV file.

W|K "W" is specified when only the window status, including the dialog box color setting, is to be saved and "K" is specified if saving softkey definitions alone. If none is specified, both window status and softkey definitions will be saved.

To retrieve the saved window status and softkey definitions from USD3.SAV file, use URECall command (Section 4.1.23).

The latest saved status in the USD3.SAV file (if any) will become the default window status and softkey definitions the next time USD-III is invoked, otherwise the original USD-III default status will be recalled.

The same command may be also executed from "OPTIONS" pull-down menu (Section 5.2.7.4.)

Chapter 5

USD-III MENU COMMANDS

MICROTEK INTERNATIONAL INC.

5.1 ABOUT THIS CHAPTER

This chapter describes each of the USD-III menu commands, from menu bar, pull-down menus, dialog boxes, down to individual fields.

The commands are grouped into 9 sections. The sections are organized and arranged according to the menu sequence in the menu bar or in the following manner:

- "File" Menu (Section 5.2.1)
- "Setup" Menu (Section 5.2.2)
- "Memory" Menu (Section 5.2.3)
- "Emulation" Menu (Section 5.2.5)
- "Trace" Menu (Section 5.2.5)
- "Symbol" Menu (Section 5.2.6)
- "Options" Menu (Section 5.2.7)
- "Spa" Menu (Section 5.2.8)
- "Lsa" Menu (Section 5.2.9)

Each of these menus will provide a pull-down menu interface (Figure 5-1). In all cases, with cursor positioned on any menu or field, pressing <F1> will display a "Help" box with instant definition for the highlighted item.

5.2 EXECUTING COMMANDS FROM MENUS

From Menu Bar:

To enter into menu command environment or move cursor to menu bar from command-line mode, press <Alt> key and the key-character (highlighted on monochrome screen or shown in red on EGA/VGA screen) of the desired menu bar menu. The pull-down menu or chart for the selected menu will immediately display.

For example, press <Alt-t> if "Trace" menu is desired. The "Trace" pull-down menu will immediately appear on the screen or right below the "Trace" menu bar menu. Likewise if "Symbol" menu is desired, press <Alt-Y> and watch the "Symbol" pull-down menu appears on the screen or right below the "Symbol" menu bar.

MENU COMMANDS

With cursor in the menu bar, $<\leftrightarrow><\rightarrow>$ may be used also (in addition to the above method) to move and select other menus in the menu bar. Observe pull-down menu appears as each menu is accessed.



Move mouse cursor to the desired Menu Bar menu and click left-button. If cursor is already inside Menu Bar, user may drag cursor along Menu Bar to the desired menu.

From Pull-Down Menu:

With keyboard, user has two options to select and execute commands from the pull-down menus:

1. Use $<\uparrow><\downarrow>$ to move and highlight the desired command from pull-down menu then press <CR>.
2. Key-in the key-character of the desired command.



Two options is also available when using mouse to execute pull-down menu commands:

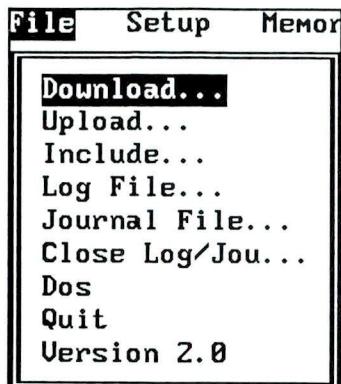
1. Drag mouse cursor across the pull-down menu, point to the desired command and release button.
2. Move cursor to the desired command and click left-button

If the selected command is suffixed with ellipses (three dots), a dialog box will appear to prompt user for command parameters before the command is executed. Otherwise, the command is executed directly from the pull-down menu.

From Dialog Box:

The dialog box will prompt user to provide the data required to complete the command selected from the pull-down menu. Refer to Sections 1.3.4 and 1.3.5.4 for keystrokes to use and mouse operation in moving around the dialog box.

5.2.1 "File" Pull Down Menu Commands



When accessed, the "File" menu will display the "FILE" pull-down menu as illustrated in Figure 5-1. This pull-down menu provides file management commands. It also offers utility for temporary exit to DOS, quitting USD-III and USD-III version display.

Execution of the "File" commands are explained in the following sections.

Note that "Dos" and "Quit" commands are directly executed from the pull-down menu for the following applications:

"**Dos**" is used to temporarily exit to DOS environment without quitting USD-III.

"**Quit**" is used to quit USD-III and return to DOS. See Sections 5.2.1.7.

"**Version**" has no function other than to display the USD-III version currently in use.

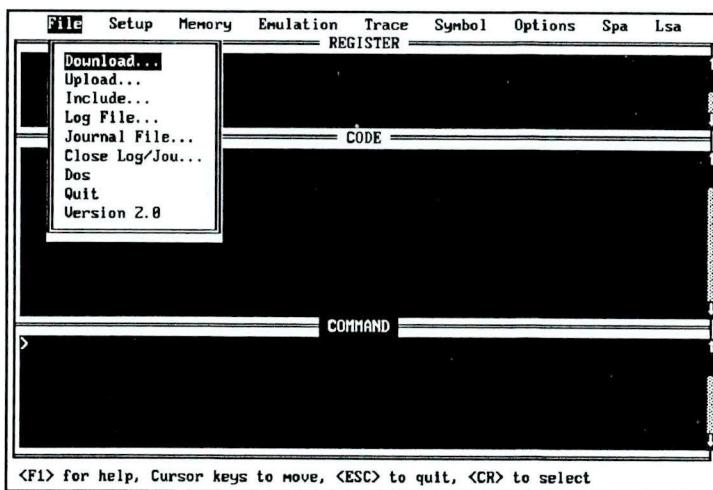


Figure 5-1 "File" Pull-Down Menu

Download...

MENU COMMAND

5.2.1.1 Download File - Download

Equivalent command syntax:

DOwnload filename [UP|UD|SP|SD|P|R]

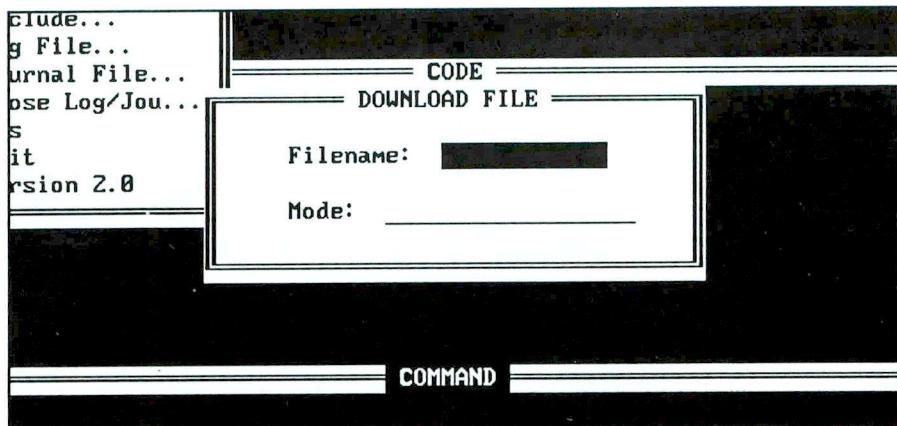


Figure 5-2 "Download File" Dialog Box

DOWNLOAD FILE dialog box is used to download an object file to MICE memory. The object file may be of an Intel hex, Intel binary (OMF), Motorola S or Tektronix hex. After downloading, the symbol table file specified in the **Filename** field is then automatically loaded.

If the entered filename is not recognized, the message-

[USD ERROR]: FILE OPEN ERROR!

will display in the COMMAND viewport.

Mode field is applicable only when USD-III is linked with MICE-16 68000/ 80286, and MICE-32 80386. This is further discussed below.

MENU COMMAND

Download...

When linked with MICE-16 68000/ 80286 and MICE-32 80386, the **Mode** field becomes accessible. Otherwise, this field is skipped. Press <F2> and the choice lists (shown below) will display. Using <**↑**><**↓**>, select the desired mode format from choice list, then press <ENTER> to execute. If none is selected, "Supervisor Program" (in the case of MICE-16 68000) or "Real Mode" (in the case of MICE-16 80286 and MICE-32 80386) will default.

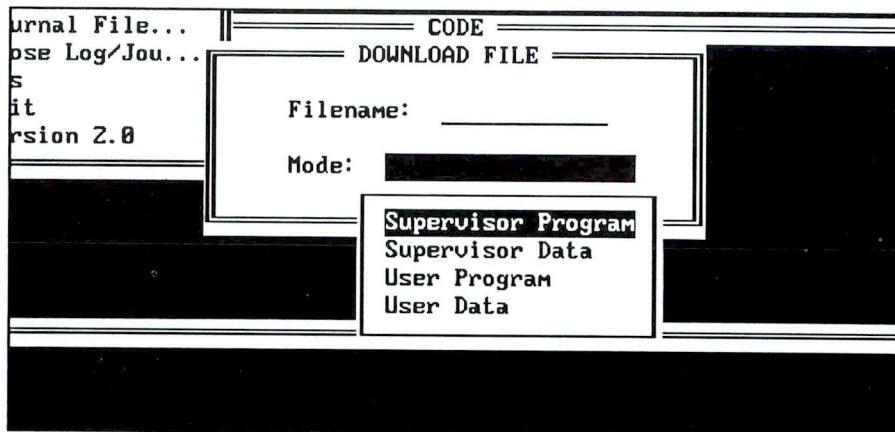


Figure 5-3a "Download Mode" Choice List (68000)

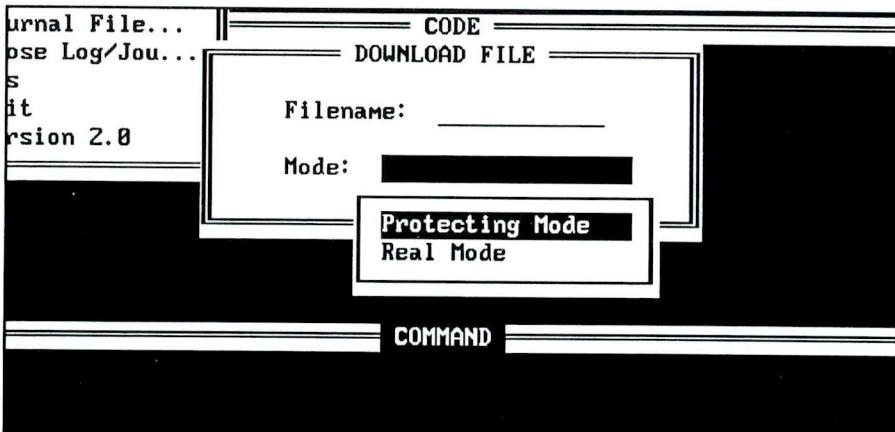


Figure 5-3b "Download Mode" Choice List (80286/386)

Download...

MENU COMMAND

If USD-III is linked with MICE other than those mentioned above, the **Mode** field is automatically skipped.

To abort downloading after it has been initiated, press <ESC> and the message-

"DOWNLOAD ABORTED!"

will display. Also note that if downloading is unsuccessful, an error message-

"DOWNLOAD FAILURE!"

will display and USD-III exits from this command.

Refer to Section 4.1.7 for the equivalent "DDownload" command syntax.

MENU COMMAND

Upload...

5.2.1.2 Upload File - Upload

Equivalent command syntax:

UUpload filename [I|[X|P]|T[X|P]|M|UP|UD|SP|SD]

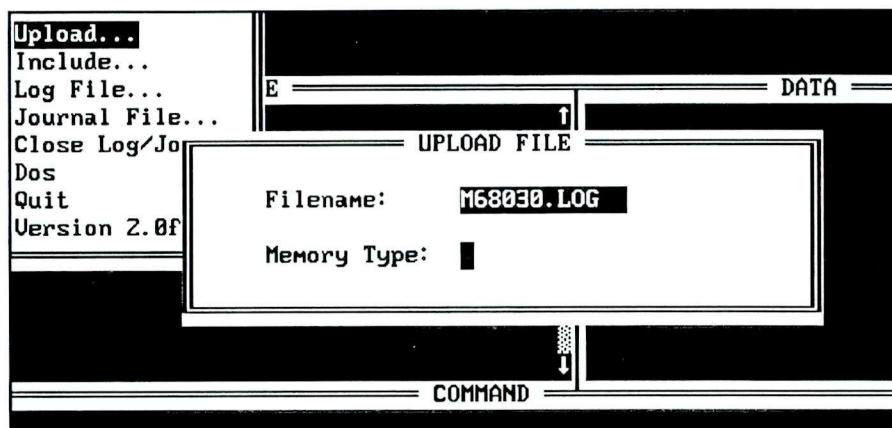


Figure 5-4 "Upload" Dialog Box

UPLOAD FILE Dialog Box uploads object code from MICE to a specified filename. Enter filename in the **Filename** field.

When linked with MICE-16 68000, MICE-32 68030 and MICE-II 8052/152, the **Memory Type** field becomes accessible. Otherwise this field is skipped. Press <F2> and the choice lists shown below will display. Using <**↑**><**↓**>, select for the desired mode format from choice list, then press <CR> to execute.

Note that when uploading the memory contents to an existing file, the file is overwritten and no warning message will be displayed.

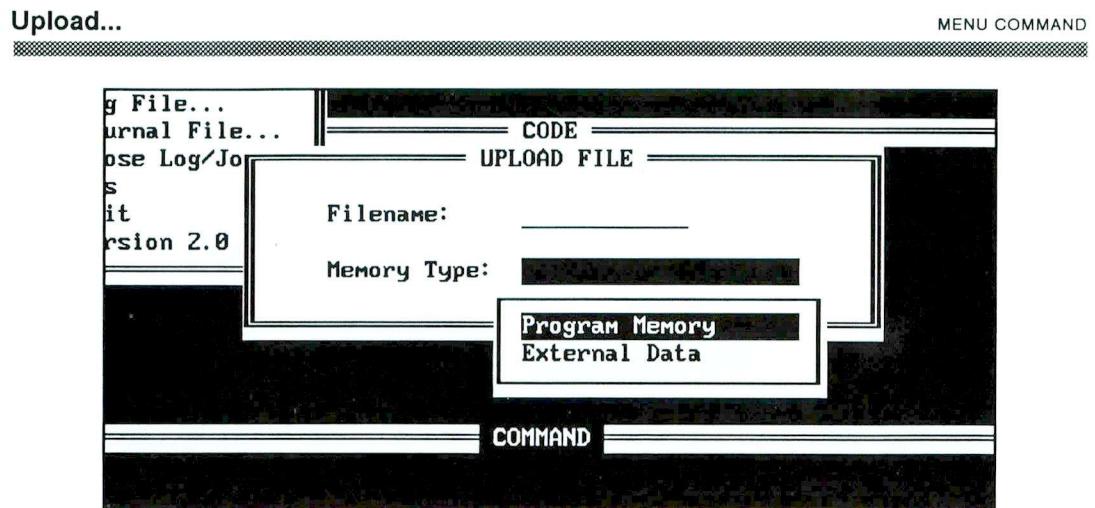


Figure 5-5a "Upload Memory Type" Choice List (5052/152)

With MICE-8 8052/ 152, If no parameter is entered, "Program Memory" will default.

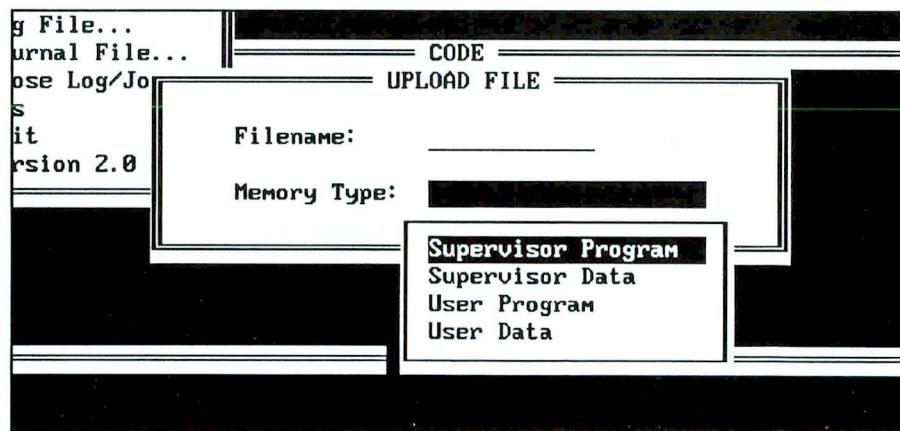


Figure 5-5b "Upload Memory Type" Choice List (68000)

With MICE-16 68000, If no parameter is entered, "Supervisor Program" will default.

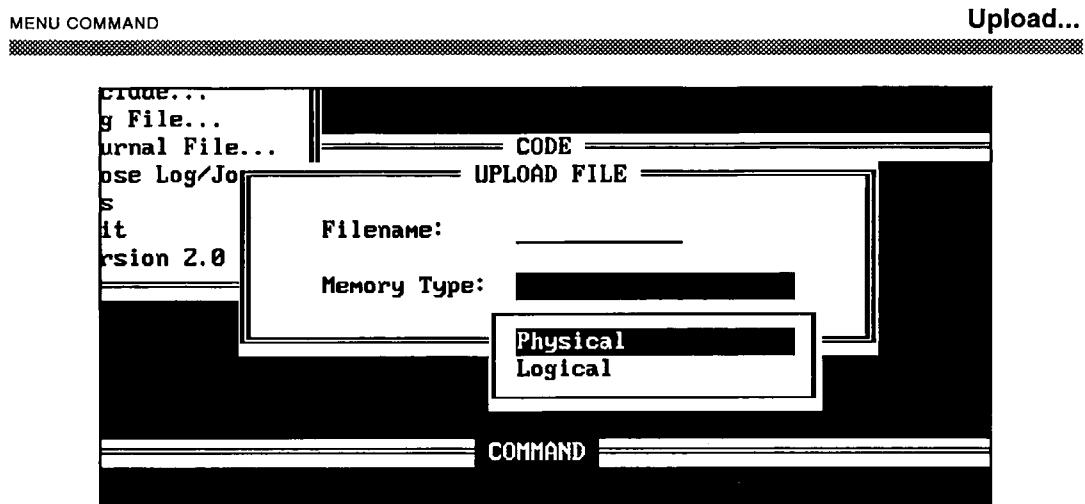


Figure 5-5c "UPLoad Mode" Choice List (68030)

With MICE-32 68030, If no parameter is entered, Logical address will default.

If the MICE model running on USD-III is of Intel 8086/88, 80186/188 or 80286, then the Code Segment value for the object code to be uploaded will display (see Example 2 of Section 4.1.23). Then the memory range request message-

"Upload memory range:"

will display. Respond with the range to be uploaded or press <ESC> to terminate upload command. More than one segment may be uploaded. The uploaded memory contents are then sent through MICE to a disk file.

If the entered filename is not recognized, the message-

[USD ERROR]: FILE OPEN ERROR!

will display.

To abort uploading after it has been initiated, press <ESC> and the message-

"UPLOAD ABORTED!"

will display.

Refer to Section 4.1.27 for the equivalent "UPload" command syntax.

Include...

MENU COMMAND

5.2.1.3 Execute Command File - Include

Equivalent command syntax:

INCLUDE filename [(arg0,...arg9)]

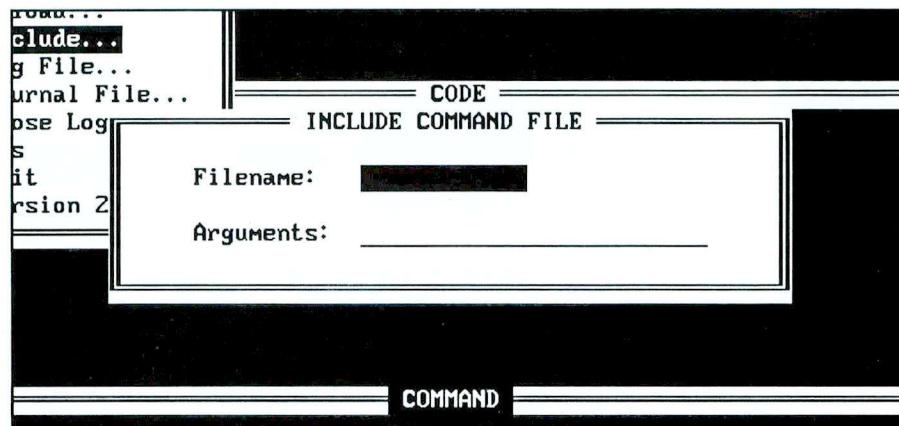


Figure 5-6 "Include" Dialog Box

INCLUDE COMMAND FILE Dialog Box is used to execute a user-defined command file. When "Include" command is executed, the variables (%0...%9) specified in the user-defined command file will be substituted with the arguments entered in this Dialog Box. Enter the user-defined command file filename in the **Filename** field.

During execution of "Include" command, if Arguments (e.g., MICE/USD-III commands, command parameters, nested commands, macros, values, etc.) are specified in the **Argument** field, the Arguments will substitute the variables (%0...%9) defined in the command file. Arguments must be entered separated by commas (NOT spaces) and enclosed in parentheses.

Example: (100,10,200,8F,0,3FFFF)

MENU COMMAND

Include...

When executed, "100" replaces "%0", "10" replaces "%1", "200" replaces "%2", etc., in the command file.

NOTE

Nested command files (see Section 4.1.11) can be specified in the argument but is limited to a maximum of one level nesting only, e.g. the nested command file should NOT contain any other 'INCLUDE' sub-command.

Refer to Section 4.1.11 and its sub-sections for the USD-III pre-defined sub-command facility for creating command files and other details.

5.2.1.4 Log Commands - Log File

Equivalent command syntax:

LOG [filename [APPEND]]

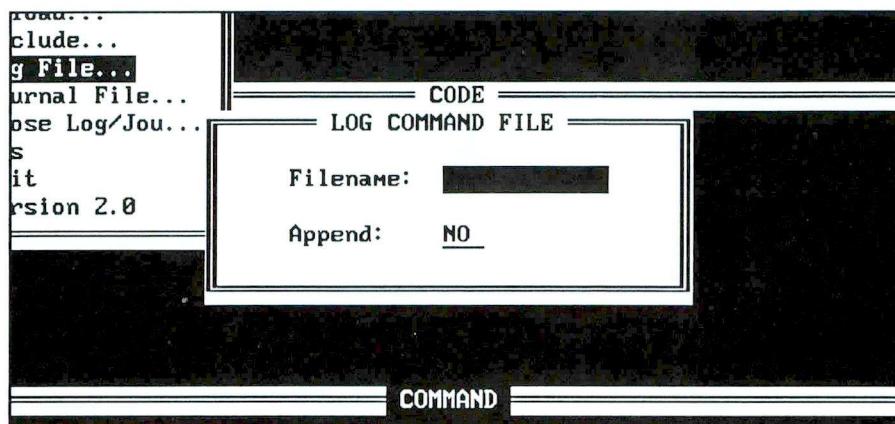


Figure 5-7 "Log Command File" Dialog Box

LOG COMMAND FILE Dialog Box is used to log all command lines to an assigned filename as they are entered and executed during a debugging session.

Assign a filename in the **Filename** field to store the logged commands lines. If it is an existing filename, instruction must be given whether to append the newly logged command lines into the existing filename or to overwrite the existing data with the newly logged data. This is done by selecting appropriate (Yes/ No) instruction with the <space-bar> **Append** toggle. If the filename does not exist, it will be created.

To intermittently disable/enable logging of executed command lines within a debugging session, refer to "Log toggle status" command in Section 5.2.7.1. Note that whenever a filename is assigned in the **Filename** field, "Log toggle status" command status is automatically toggled to "ON".

MENU COMMAND

Log File...

To save the logged command lines into the assigned filename and end logging session, refer to "Close Log/Jou" command in Section 5.2.1.6.

Refer to Section 4.1.15 for the equivalent "LOG" command syntax.

Journal File...

MENU COMMAND

5.2.1.5 Log Output and Create Output File - Journal File

Equivalent command syntax:

JOURNAL [filename [APPEND]]

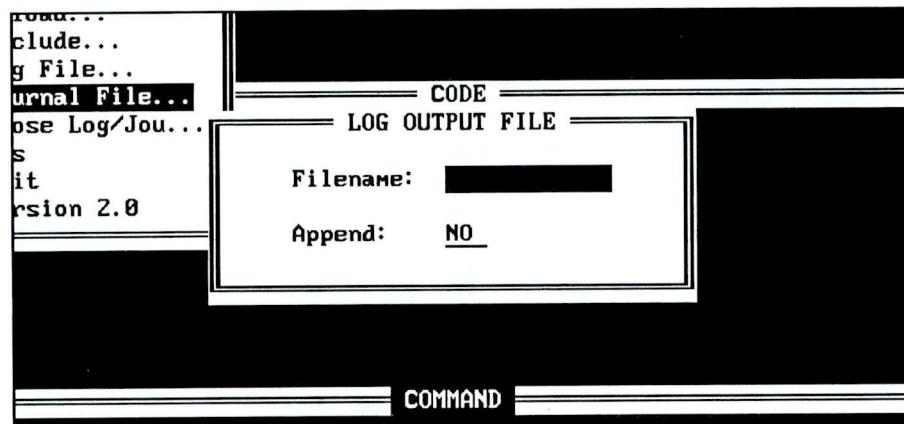


Figure 5-8 'Journal File' Dialog Box

LOG OUTPUT FILE Dialog Box logs all COMMAND viewport activities (input and responses) to a designated filename as they took place during a debugging session.

Enter a filename to store the logged COMMAND viewport activities in the **Filename** field. If it is an existing filename, instruction must be given whether to append the newly logged command lines into the existing filename or to overwrite the existing data with the newly logged data. This is done by selecting appropriate (Yes/ No) instruction with the <space-bar> **Append** toggle. If the filename does not exist, it will be created.

To intermittently disable/ enable logging of executed command lines within a debugging session, refer to "Journal toggle status" command in Section 5.2.7.1. Note that whenever a filename is assigned in the **Filename** field, "Journal toggle status" command status is automatically toggled to "ON".

MENU COMMAND

Journal File...

To save the logged COMMAND viewport activities into the assigned filename and end logging session, refer to "Close Log/ Jou" command in Section 5.2.1.6. Watch "JO CLOSE" being displayed in COMMAND viewport.

Refer to Section 4.1.13 for the equivalent "JOurnal" command syntax.

Close Log/Jou...

MENU COMMAND

5.2.1.6 Close Log/Journal File - Close Log/Jou

Equivalent command syntax:

LOG|JOURNAL CLOSE

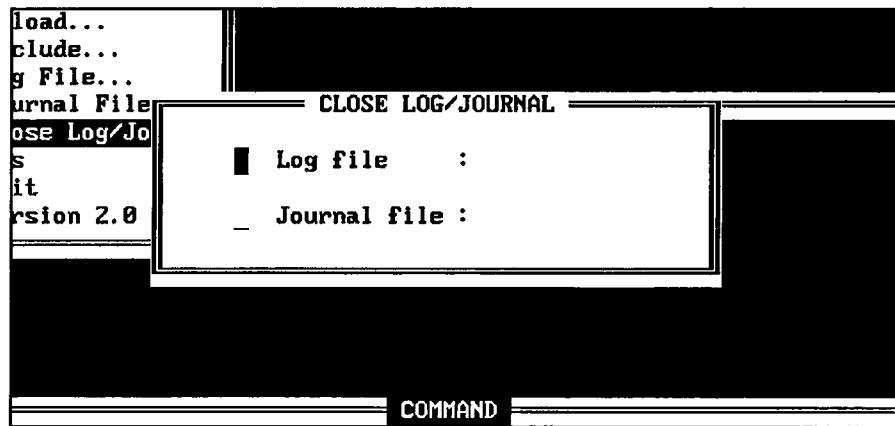


Figure 5-9 "Close Log/Journal" Dialog Box

CLOSE LOG/JOURNAL dialog box is used to save and close the current log or journal file session, or both. A log or journal filename should be assigned before-hand (see Sections 5.2.1.4 and 5.2.1.5) before this feature could be carried out properly. Such filename should show up on their corresponding field. Otherwise the field will be empty.

With cursor on **Log file** field, <space-bar> toggle to enable/disable "save and close" log file feature. Check mark (✓) indicates feature is enabled. Repeat the same steps for **Journal file** field.

Refer to Section 4.1.13 and 4.1.15 for the equivalent "JOURNAL" and "LOG" command syntax respectively.

5.2.1.7 Quit USD-III - Quit

Equivalent command syntax:

QUIT

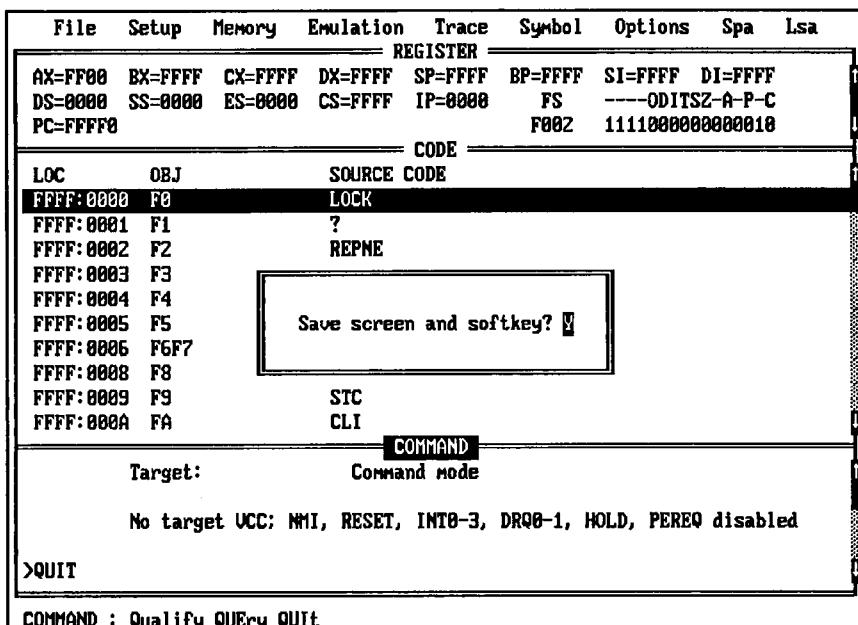


Figure 5-10 "Quit USD-III" Dialog Box

After executing the **Quit** command from the pull-down menu, a dialog-box will display to confirm whether user wishes to **"Save screen and softkey?"**. If the answer is "Y", USD-III will execute the **"USAve"** command (Sections 5.2.7.4) internally before exiting. If the answer is "N", the system will immediately return to MS-DOS.

Refer to Section 4.1.21 for the equivalent **"QUIT"** command syntax.

"Setup" Pull-Down Menu

MENU COMMAND

5.2.2 "Setup" Pull Down Menu Commands



The "SETUP" pull-down menu provides commands for memory mapping as illustrated in the Figure 5-11. It also recalls and saves MICE setup parameters from and to NOVRAM. Execution of these commands are explained in the following section.

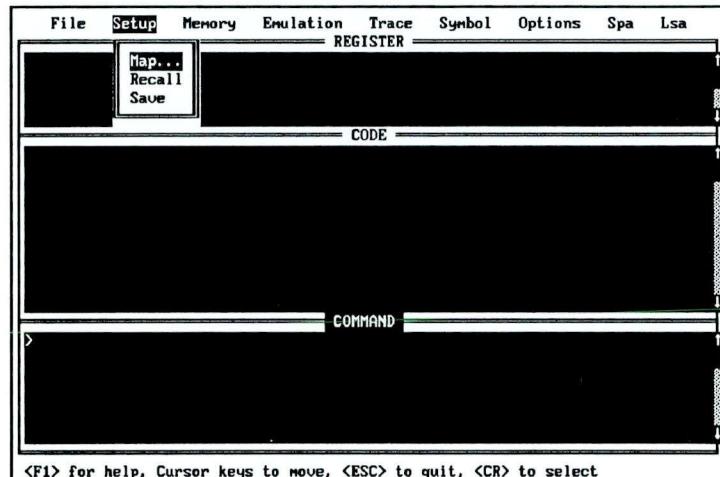


Figure 5-11 "SETUP" Pull-Down Menu

MENU COMMAND

Map...

5.2.2.1 Memory Map - Map

Equivalent line command syntax:

refer to applicable MICE manual

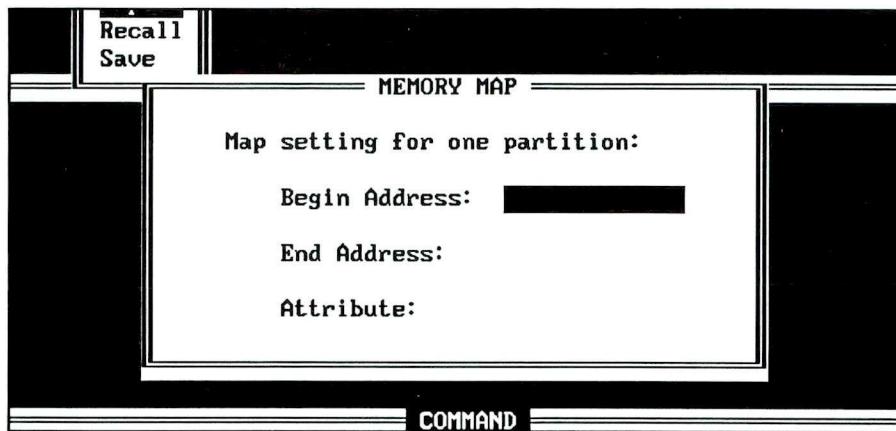


Figure 5-12 "Memory Map" Dialog Box

MEMORY MAP dialog box maps a defined block of memory. Define the starting and the end address of the block to be mapped in the **Begin Address** and **End Address** fields. If no memory block is defined, the current map setting will display.

If attributes are required for the defined block of memory for mapping, enter the attributes in the **Map Attribute** field.

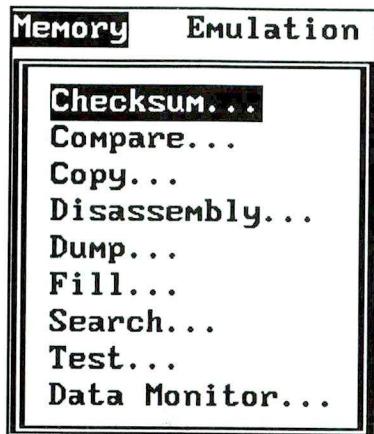
To save the current emulation memory map settings and other data to the non-volatile RAM (NOVRAM), return to the "Setup" pull-down menu and execute "Save" command. Watch a list of categories of the data (not the data itself) saved in the NOVRAM being displayed in the COMMAND viewport. Also refer to applicable MICE manual for other important details on MICE "Save" command.

Map...

MENU COMMAND

To recall the emulation memory map and timebase, clock, channel B baud rate and select code currently saved in the non-volatile RAM (NOVRAM), return to "Setup" pull-down menu. Invoke "Recall" command and watch a list of categories of the data (not the data itself) recalled from NOVRAM being displayed in the COMMAND viewport. Note that this command is automatically executed at MICE power-up or after a hardware reset. Also refer to applicable MICE manual for other important details on the MICE "Recall" command.

5.2.3 "Memory" Pull-Down Menu Commands



The "MEMORY" pull-down menu provides access to the contents of designated memory locations as illustrated in Figure 5-13. Execution of "Memory" commands are explained in the following sections.

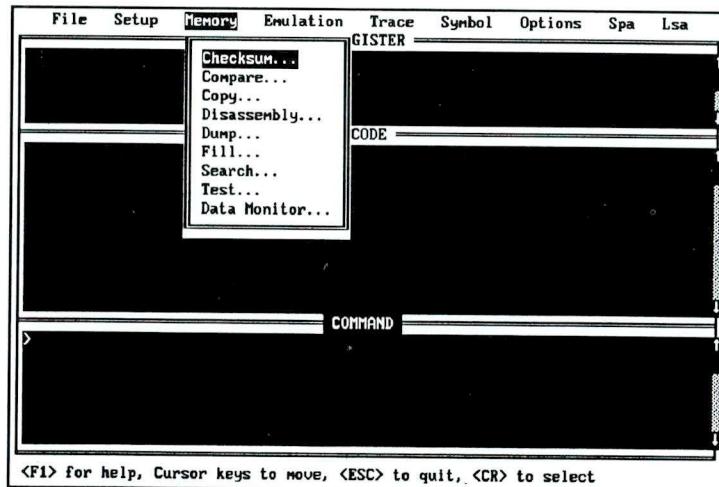


Figure 5-13 "Memory" Pull-Down Menu

Checksum...

MENU COMMAND

5.2.3.1 Memory Checksum - Checksum

Equivalent command syntax:

refer to applicable MICE manual

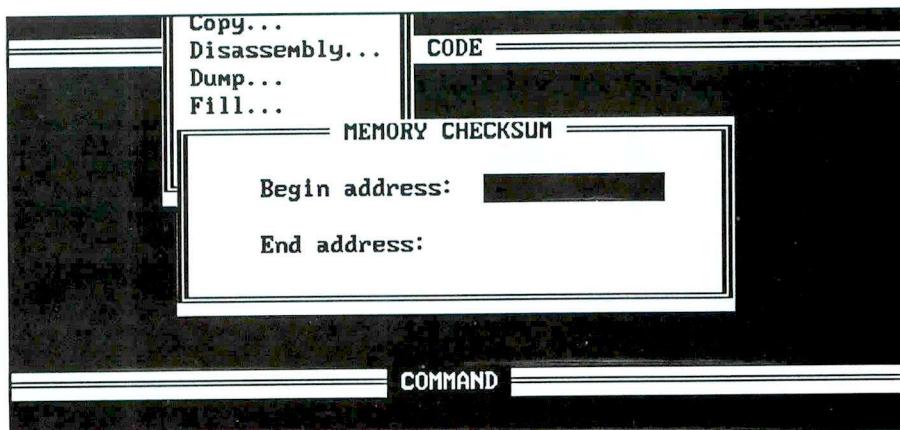


Figure 5-14 "Memory Checksum" Dialog Box

MEMORY CHECKSUM dialog box will accomplish checksum on the memory range between the specified begin address and end address.

Key-in the memory location where the checksum operation is to begin in the **Begin address** field. Move to the **End address** field and enter the last memory location where checksum is to be carried out.

Refer to MICE manual for more details.

MENU COMMAND

Compare...

5.2.3.2 Memory Compare - Compare

Equivalent command syntax:

refer to applicable MICE manual

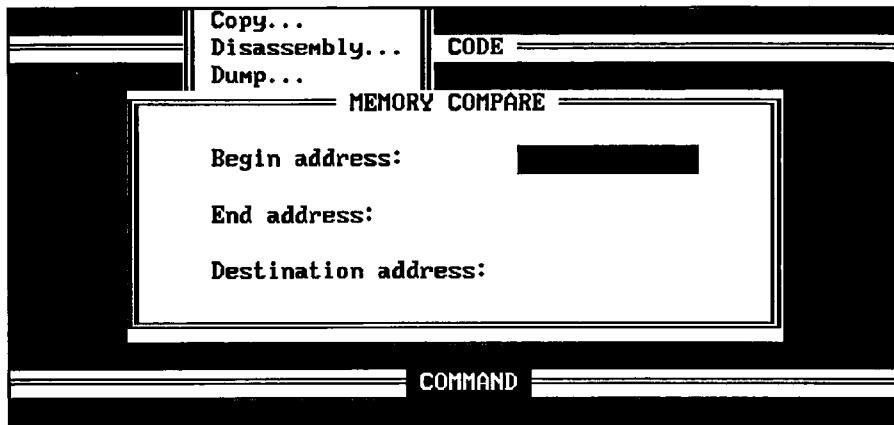


Figure 5-15 "Memory Compare" Dialog Box

MEMORY COMPARE dialog box compares two defined blocks of memory when the start and end address of the first block are specified. Only the start address is needed to be specified for the second block.

Specify the starting address of block 1, the first of the two memory ranges to be compared, in the **Begin address** field. Move to the **End address** field and enter the last location in block 1. Specify the starting point of block 2 in the **Destination address** field.

Refer to MICE manual for more details.

Copy...

MENU COMMAND

5.2.3.3 Memory Copy - Copy

Equivalent command syntax:

refer to applicable MICE manual

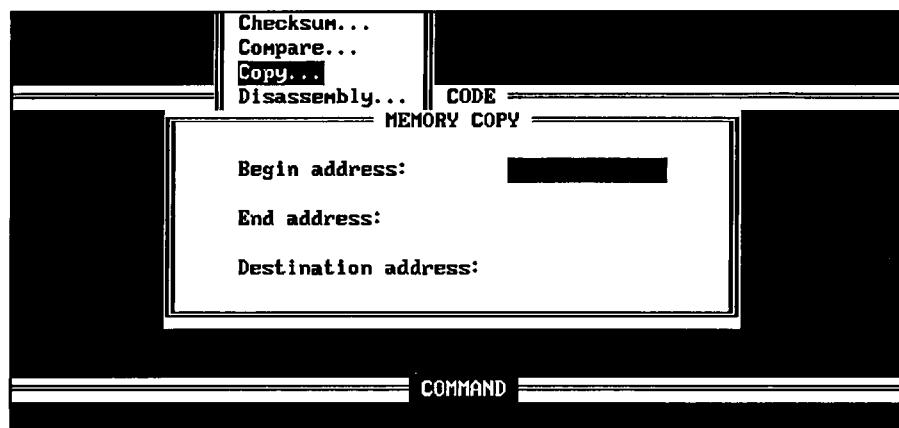


Figure 5-16 "Memory Copy" Dialog Box

MEMORY COPY dialog box is used to copy the content of a memory block to another within emulation or target memory. It is also used for copying a memory block from target memory to emulation memory.

Specify the first location in the memory range to be copied in the **Begin address** field. Move to the **End address** field and enter the last location in the memory range to be copied. Enter the starting address of the memory range to which the specified data are to be written in the **Destination address** field.

Refer to MICE manual for more details.

5.2.3.4 Memory Disassembly - Disassembly

Equivalent command syntax:

refer to applicable MICE manual

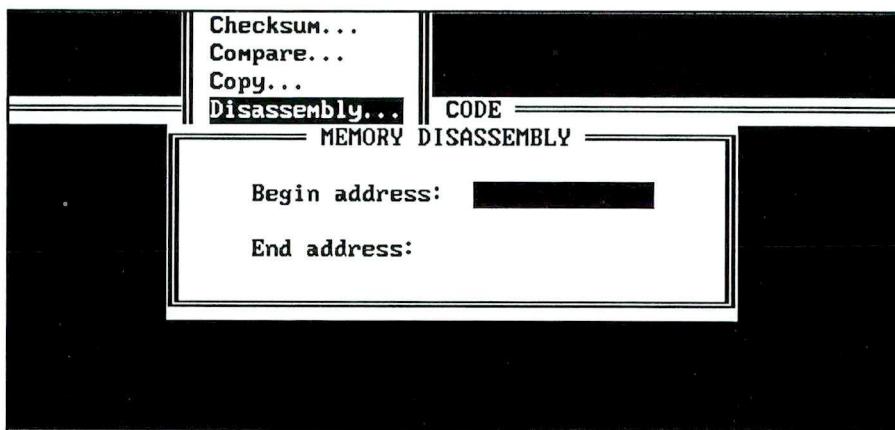


Figure 5-17 "Memory Disassembly" Dialog Box

MEMORY DISASSEMBLY dialog box disassembles and displays the contents of the memory, one page (16 statement line) at a time. The starting and end address of the memory contents must be specified, otherwise disassembly will start at the current program counter (pc).

Specify the first memory location where the first instruction to be disassembled resides, in the **Begin address** field. Move to the next **End address** field and enter the last location of the memory range to be disassembled.

Refer to MICE manual for more details.

Dump...

MENU COMMAND

5.2.3.5 Memory Dump - Dump

Equivalent command syntax:

refer to applicable MICE manual

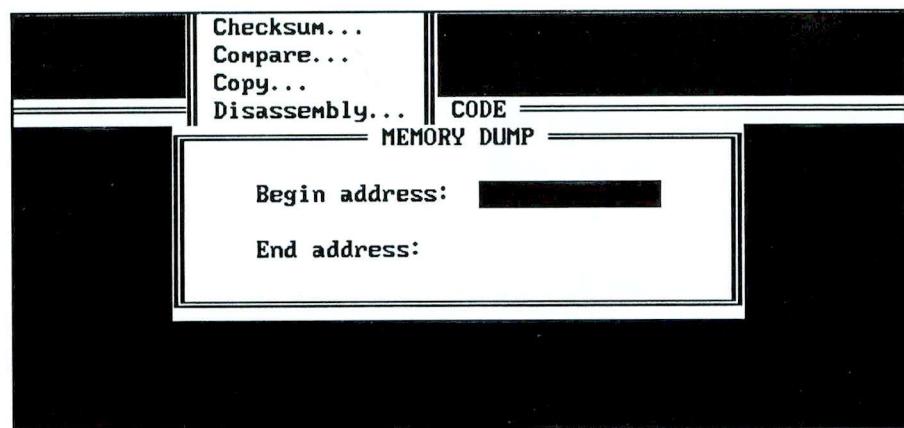


Figure 5-18 "Memory Dump" Dialog Box

MEMORY DUMP dialog box will display the contents of a specified block of memory. If no address is specified, memory display will start at the current program counter (pc). If the starting address is specified and no end address is given, the contents of memory will display one page (256 bytes) at a time. The message "[MORE]" will appear at the bottom of each page. Press <CR> to view the next page.

Specify the memory location where the memory dump operation is to begin in the **Begin address** field area. Move to the **End address** field and enter the last memory location to be dump.

Refer to MICE manual for more details.

MENU COMMAND

Fill...

5.2.3.6 Memory Fill - Fill

Equivalent command syntax:

refer to applicable MICE manual

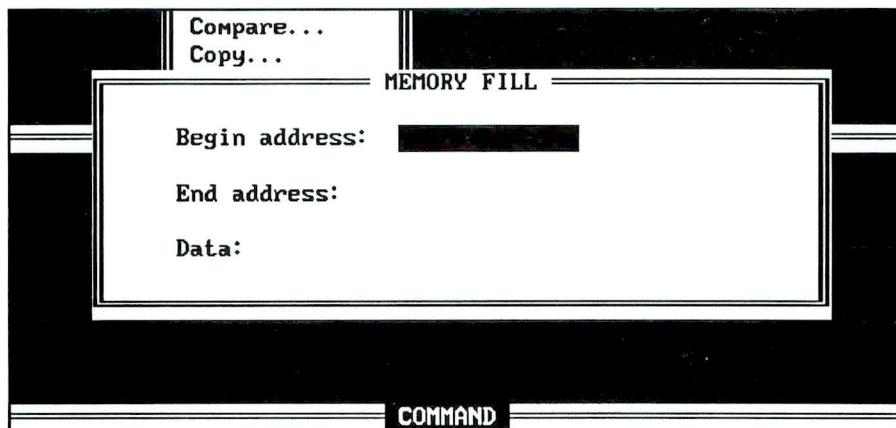


Figure 5-19 "Memory Fill" Dialog Box

MEMORY FILL dialog box fills a specified memory block with a particular value or string.

Specify the memory location where the fill operation is to begin in the **Begin address** field area. Move to the **End address** field and enter the last location in the memory range to be filled. Input string in ASCII or hex in the **Data** field.

Refer to MICE manual for more details.

Search...

MENU COMMAND

5.2.3.7 Memory Search - Search

Equivalent command syntax:

refer to applicable MICE manual

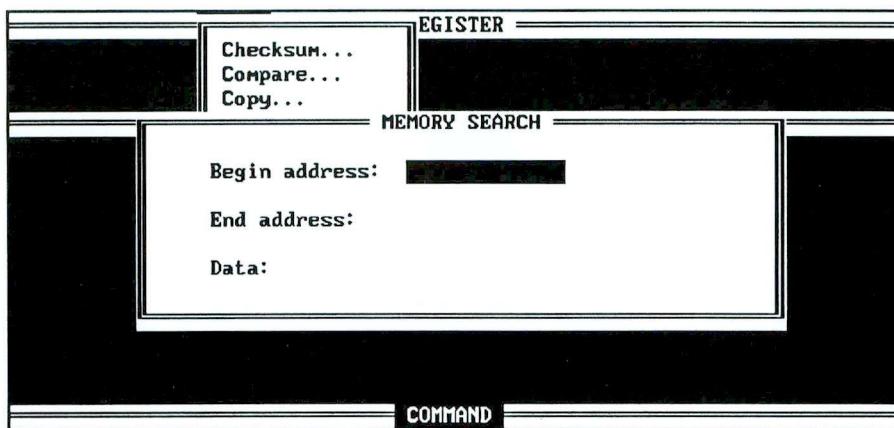


Figure 5-20 "Memory Search" Dialog Box

MEMORY SEARCH dialog box searches a particular value or string within the boundary of a specified memory block.

Specify the memory location where the search operation is to begin in the **Begin address** field area. Move to the next **End address** field and enter the last memory location in the memory range to be searched. Input string in ASCII or hex in the **Data** field.

Refer to MICE manual for more details.

MENU COMMAND

Test...

5.2.3.8 Memory Test - Test

Equivalent command syntax:

refer to applicable MICE manual

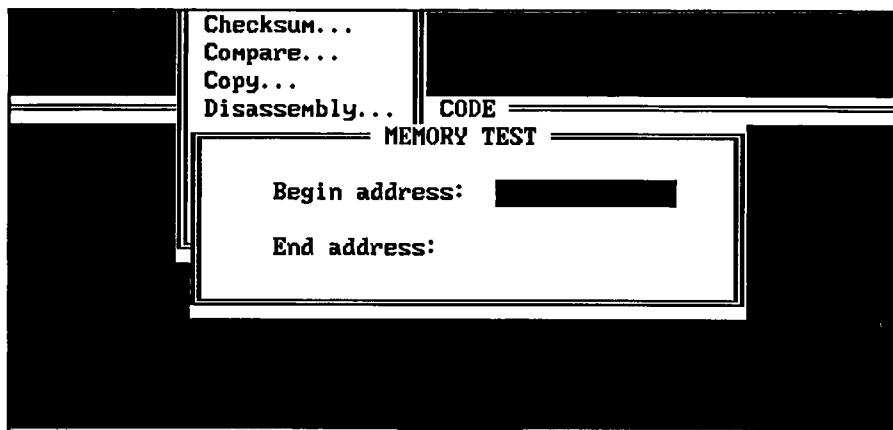


Figure 5-21 "Memory Test" Dialog Box

MEMORY TEST dialog box performs test operation on each location of a specified memory block.

Specify the memory location in the memory range where the testing operation is to begin in the **Begin address** field area. Move to the **End address** field and enter the last location in the memory range to be tested.

Refer to MICE manual for more details.

5.2.3.9 Set/Clear Monitor Variable - Data Monitor

Equivalent command syntax:

```
MONitor {[[%module]%symbol|hex_address] [number] [Byte|Word|Long]}
NOMon [no1 [no2]]
```

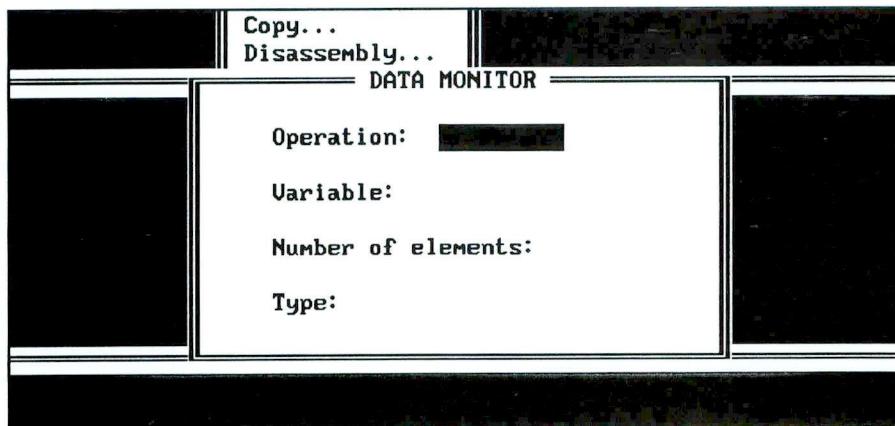


Figure 5-22 "Data Monitor" Dialog Box

DATA MONITOR dialog box is used to set variables to be displayed and monitored on the DATA viewport. The same dialog box is used to clear such variables from viewport.

The **Operation** field receives instruction to "SET", "CLEAR" part of, or "CLEAR ALL" of the variables currently on monitor. The instructions may be entered directly or press <F2> to display the choice list shown in Figure 5-23 with cursor on **Operation** field.

When "SET" is administered, cursor will prompt at the subsequent fields in the dialog box for data monitor variables specification, where:

Variable is the variables (symbol or address) to be monitored and displayed on the DATA viewport. Module name in "[[%module]%symbol" symbol format may also be specified, but only the specified module will be searched.

MENU COMMAND

Data Monitor...

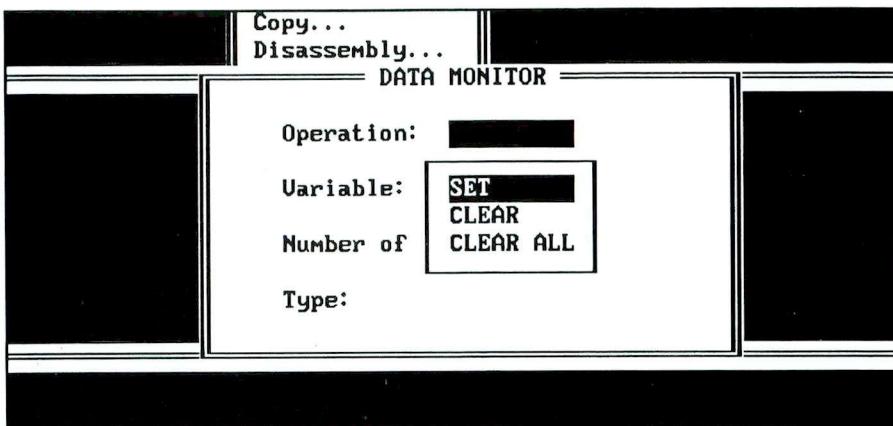


Figure 5-23 "Operation" Choice List

Number of Elements is the number of elements (filled in as a numeric number) to be displayed beginning at the specified symbol or address to be monitored. These elements are numbered as +1, +2, +3, etc., on the Data viewport (Figure 5-24). The unit of each element is defined as either Byte, Word or Long type (see next argument).

| DATA | | |
|------|------|---------------|
| 1. | 120A | = FFFF |
| | 120A | +1 = FFFF |
| | 120A | +2 = FFFF |
| 2. | 13F0 | = FFFFFFFF |
| | 13F0 | +1 = FFFFFFFF |
| 3. | 25F0 | = FF |
| | 25F0 | +1 = FF |
| | 25F0 | +2 = FF |
| | 25F0 | +3 = FF |
| | 25F0 | +4 = FF |

Figure 5-24 DATA Viewport Display

Data Monitor...

MENU COMMAND

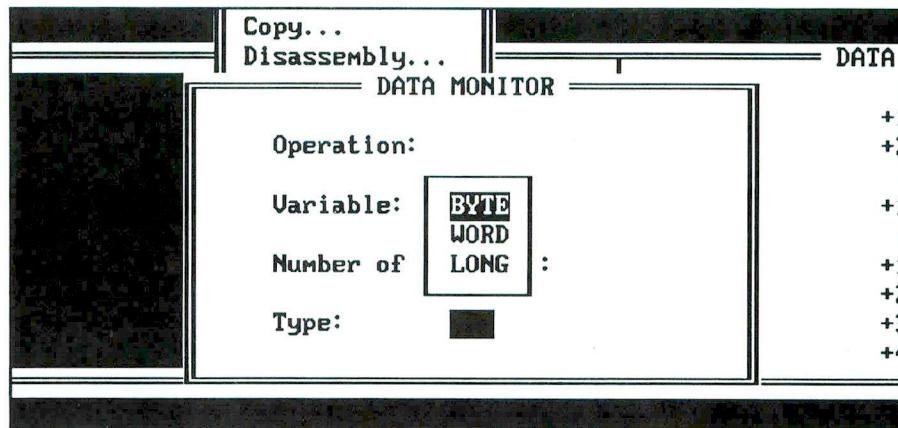


Figure 5-25 "Type" Choice List

Type is the unit for the elements of the specified variable. The type field is filled in with either Byte, Word or Long. The default is Byte. Type may also be selected from the Type choice list (Figure 2-25) by pressing <F2> with cursor on Type field.

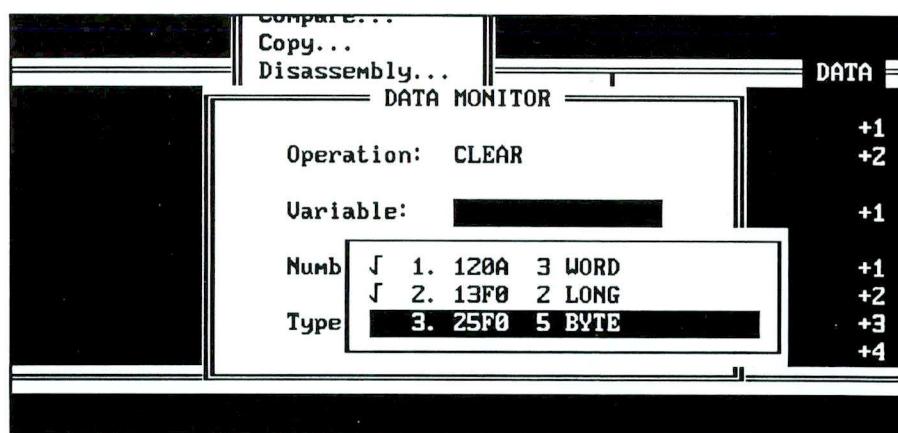


Figure 5-26 "Clear" Choice List

When "CLEAR" is entered on the Operation field or selected from the choice list, another choice list (Figure 5-26) will pop-up to display all the monitored variables. Move cursor to the variable to be cleared and press <Enter> to toggle a check mark (✓) prefix on the variable. Repeat the same routine to other variables to be deleted. Press <F10> to confirm selection, and a second <F10> to execute.



A click on any item in the choice list will toggle check mark (✓). Another click with cursor on the border of the choice list will save the selection.

To clear all existing variables, enter or select "CLEAR ALL". Execute with either <ENTER> or <F10>.



Click left-button with cursor on any area within the dialog box other than the fields.

Refer to Section 4.1.17 for the equivalent "MONitor" command syntax.

"Emulation" Pull-Down Menu

MENU COMMAND

5.2.4 "Emulation" Pull-Down Menu Commands

Equivalent command syntax:

IStep [Over] [count] | FForever



When pulled down, the "EMULATION" pull-down menu will display components of instruction step command and cycle step command as illustrated in Figure 5-27. These commands provide no dialog boxes and are executed directly from the pull-down menu. Note that "Istep Over" and "Istep Over Forever" commands are applicable to MICE-III only. The commands are skipped if USD-III is linked to MICE-II.

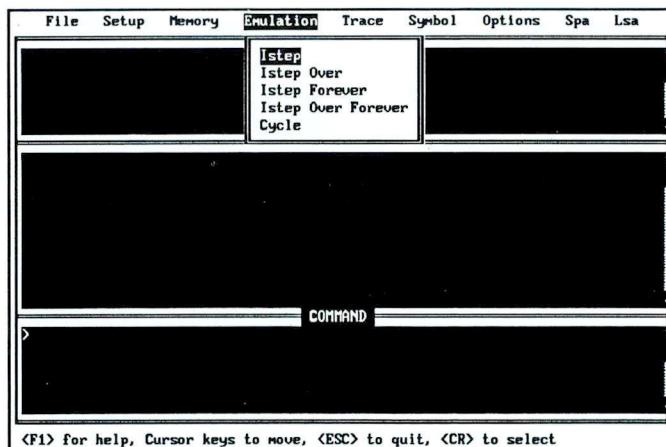


Figure 5-27 "Emulation" Pull-Down Menu

Istep command will execute instruction by single-stepping and returns to command prompt mode after executing the command. After each step, the REGISTER, CODE, DATA and STACK viewports information are also updated accordingly if their respective updating toggle is enabled.

- Istep Over** is a MICE-III instruction step command option to treat subroutines as normal instructions. With this option, user needs NOT step into subroutines which he is not concerned about. USD-III will run through such subroutines.
- Istep Forever** is an instruction step command option to step instruction continuously until forced to stop by pressing <Esc> key.
- Istep Over Forever** is a MICE-III instruction step command option to treat subroutines as normal instructions and to step instruction continuously until forced to stop by pressing <Esc> key. With this option, user needs NOT step into subroutines which he is not concerned about. USD-III will run through such subroutines in real time.
- Cycle** is a cycle step command to stop the emulation processor, steps the program one cycle, and then halts the processor in HOLD state.

"Trace" Pull-Down Menu

MENU COMMAND

5.2.5 "Trace" Pull-Down Menu Commands



When pulled down, the "TRACE" pull-down menu will display commands for trace program execution as illustrated in Figure 5-28. Execution of these menu commands are explained in the following sections.

Note that "Trace Control" works only with MICE-III, while "Backward/ Forward Trace" are applicable to MICE-II only.

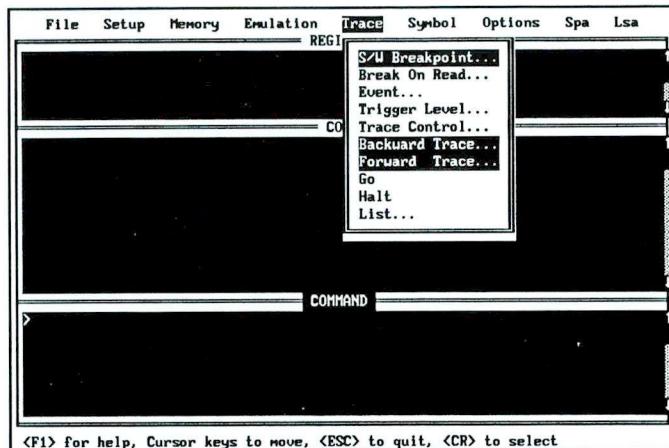


Figure 5-28 "Trace" Pull-Down Menu

"Go" and "Halt" commands are directly executed from the pull-down menu with following applications"

"**Go**" is the MICE command to start realtime emulation of the emulation processor.
"**Halt**" is the MICE command to stop program execution immediately.

MENU COMMAND

S/W Breakpoint...

5.2.5.1 Set/Clear Software Breakpoint - S/W Breakpoint

Equivalent command syntax:

BREAKPOINT [address]
CLEAR [no1 [no2]]

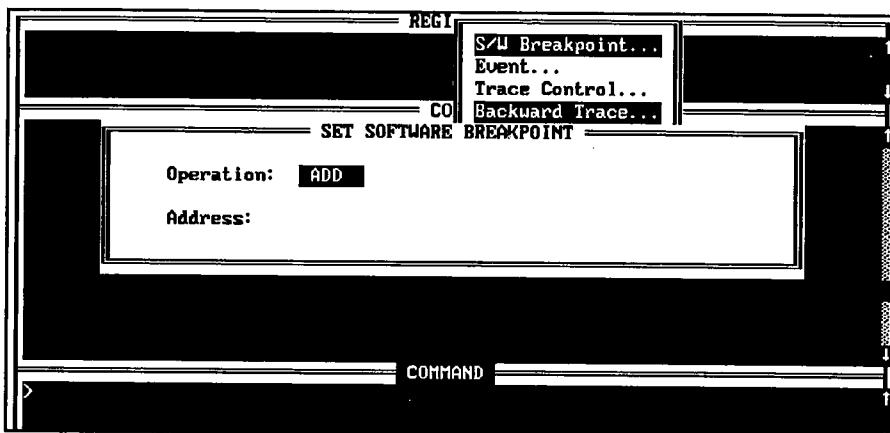


Figure 5-29 'S/W Breakpoint' Dialog Box

SET SOFTWARE BREAKPOINT dialog box is used to set or clear a software execution breakpoint.

With cursor on **Operation** field, <space-bar> toggle between "ADD" and "DELETE" to add and clear breakpoints respectively. With cursor on **address**, press <F2> to display a choice list (Figure 5-30) showing current breakpoint settings if any. If none is available, "**No choice available**" message will appear at the bottom (message line) of the screen.

To add a breakpoint, toggle to "ADD" then move cursor to **address** field and type-in the new setting.

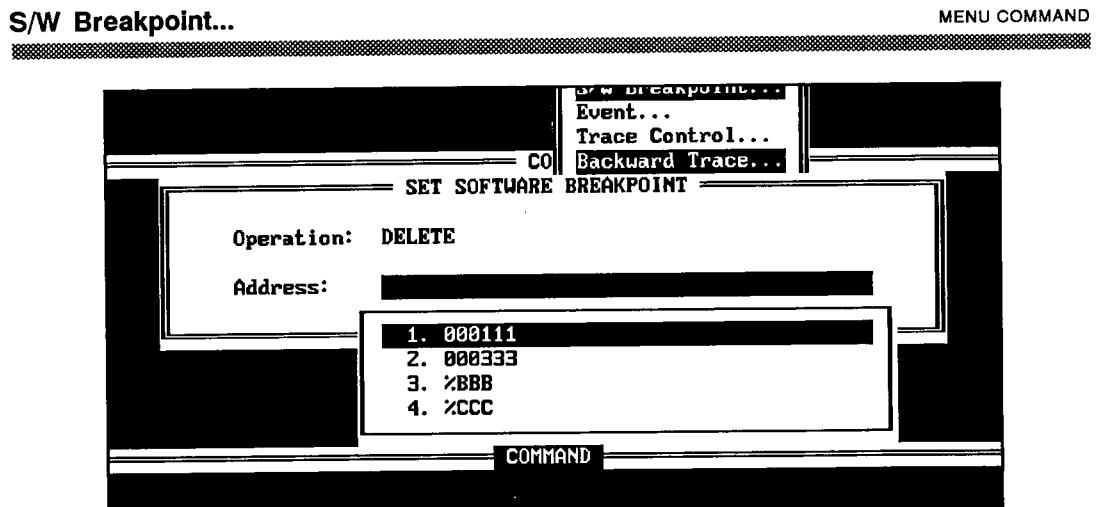


Figure 5-30 "Breakpoint Address" Choice List

To "DELETE", enter the address to be deleted in the **address** field, or display choice list (by pressing <F2>), then highlight the address to be deleted and press <CR>.

Program execution will halt whenever the breakpoint setting is match. To execute continuously without stopping at breakpoints, set "S/W Breakpoint toggle status" command (Section 5.2.7.1) to "OFF".

Refer to Section 4.1.1 for the equivalent "BREakpoint" command syntax.

MENU COMMAND

Break On Read...

5.2.5.2 Break on Read Before Write - Break On Read (for LAM-II only)

Equivalent command syntax:

refer to applicable MICE-III manual and look for INitialize command

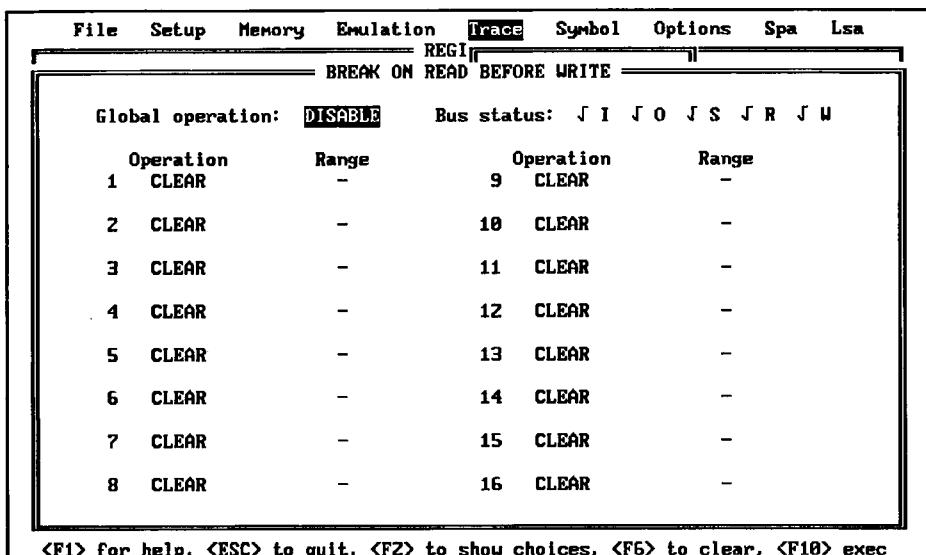


Figure 5-31 "Break On Read" Dialog Box (80C186)

BREAK ON READ dialog box provides the features offered by the INitialize command of LAM-II equipped MICE-III. It is used to ensure that only memory range which were previously written, can be read during emulation. This function is most helpful when emulating such areas as vector space, input/output, RAM, etc.

Break On Read...

MENU COMMAND

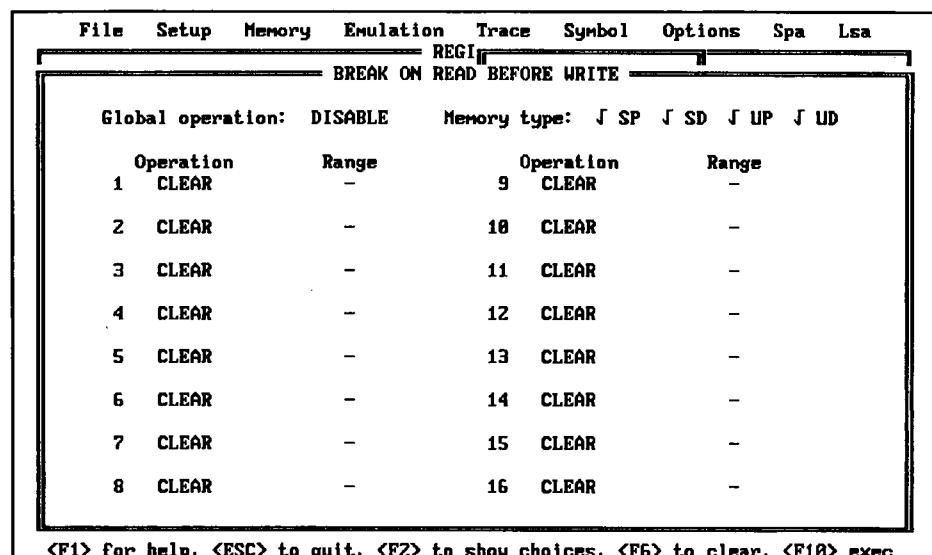


Figure 5-32 "Break On Read" Dialog Box (68K/68020/68030)

The dialog box pops up with the current MICE settings displayed on each field. To set, clear or change settings, move cursor to **Global Operation** field and enter desired operation, i.e., *SET*, *RESET*, *CLEAR*, *ENABLE* and *DISABLE* or press *<F2>* to display a choice list shown in Figure 5-33. The choices operates as follows:

SET sets memory ranges with following steps:

1. Use *<↑><↓>* or *<Enter>* to move into the **Bus Status (Memory Types** for 68K/ 68020/ 68030) field where bus status are displayed. Move cursor on the bus status to be enabled and press *<space-bar>* to toggle a check mark (*✓*) prefix on it. Repeat the same routine to other bus status to be enabled, if any.
2. Then proceed to move cursor into appropriate field under **Operation**. Use *<space-bar>* again to toggle and make selection between *CLEAR* and *SET*. When *CLEAR* is selected, existing memory blocks corresponding to the *CLEAR* fields are cleared after pressing *<F10>*.

MENU COMMAND

Break On Read...

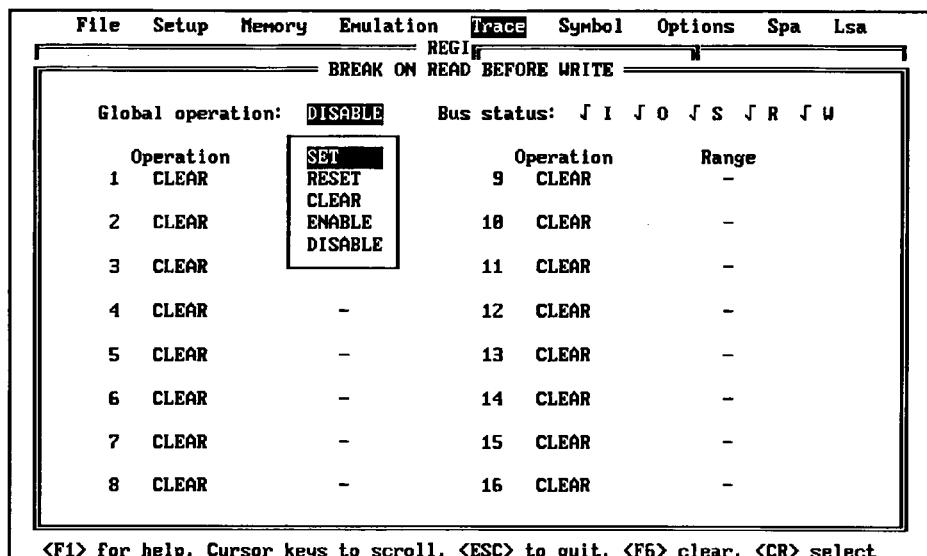


Figure 5-33 "Global Operation" Choice List (80C186)

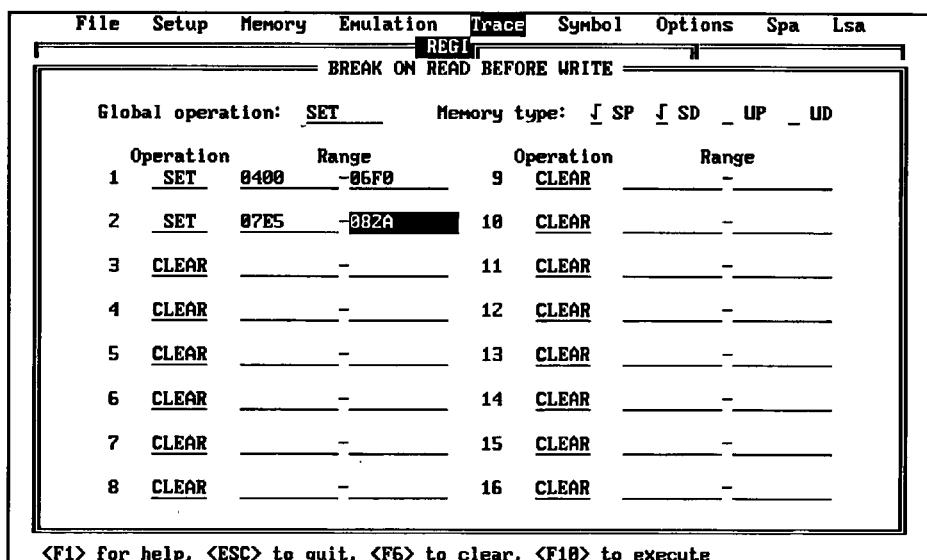


Figure 5-34 Setting Memory Block Range in 'Range' Field
(68K/68020/68030)

Break On Read...

MENU COMMAND

3. If SET Operation is selected, move cursor to the appropriate field under Range. Enter the starting point of the memory block range (in hexadecimal address) to be "initialized". Then continue to move to the next (right column) field and enter the last point of the memory block (Figure 5-34). Repeat procedure until all desired memory block ranges are set. Press <F10> to execute.

NOTE

Up to 20 initialized ranges of memory may be set in four independent 256K-byte of memory segments for this function. However, the dialog box could only accommodate 16. Use INItialize command if the full 20 memory block range settings are required.

RESET clears the results from last run without clearing the address ranges setting. Press <Enter> to execute.

CLEAR reset and clears all existing address settings and all results from last run. Press <Enter> to execute.

ENABLE is a global control to enable the "Break on Read" function. Press <Enter> to execute.

NOTE

If the "Coverage" (Section 5.2.8.4) command is active at the time when the "Break On Read" (INItialize) command is enabled, the "Coverage" command is automatically switched to inactive condition. Likewise, "Break On Read" function is disabled when "Coverage" command is enabled while "Break On Read" is active. Hence only one command ("Break On Read" or "Coverage") is active at a time.

DISABLE is a global control to disable the "Break on Read" function. Press <Enter> to execute.

If "Break On Read" is enabled while "Coverage" command is active, the latter is disabled and the following message will display:

"READ-BEFORE-WRITE is globally disabled"

Refer to MICE-III manuals for more details.

MENU COMMAND

Event...

5.2.5.3 Display/Set/Clear Breakpoints - Event

Equivalent command syntax:

refer to applicable MICE manual

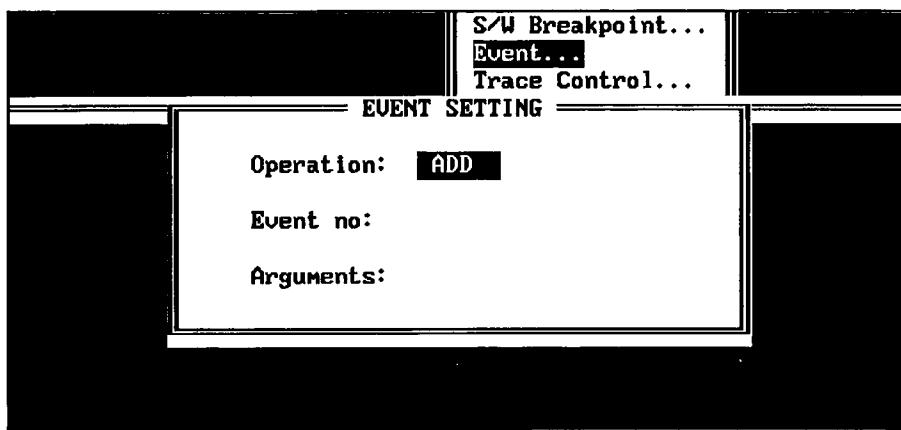


Figure 5-35 "Event Setting" Dialog Box

EVENT SETTING dialog box displays, adds and deletes event settings.

With cursor on **Operation** field, <space-bar> toggle between "ADD" and "DELETE" to add and clear event setting respectively. Enter the event number to be added/ deleted in the **Event no.** field. Valid event numbers are 1 to 6 for LAM, and 1 to 8 for LAM-II. Observe the corresponding MICE event setting displays in the **Arguments** field. If event setting is cleared, this field becomes empty.

Refer to MICE manual for more details.

Trigger Level...

MENU COMMAND

5.2.5.4 Set Trigger Level - Trigger Level (for LAM-II only)

Equivalent command syntax:

refer to applicable MICE-III manual

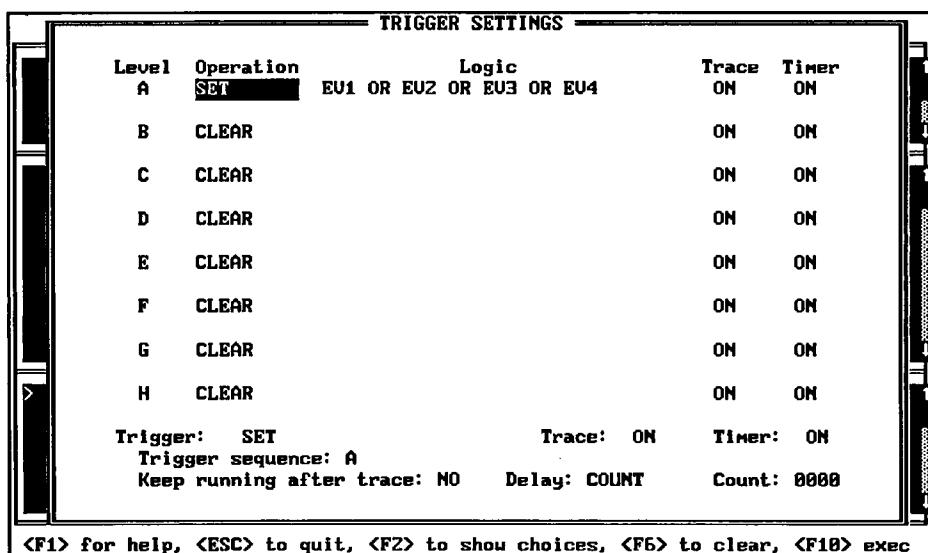


Figure 5-36 "Trigger Level" Dialog Box

TRIGGER SETTINGS dialog box sets 1 to 8 trigger levels, sets trigger settings and executes "Trigger" command. The dialog box defaults with current MICE trigger settings. The settings may be partially or wholly cleared or modified under this dialog box.

The fields under **Operation** are used to enter desired operation i.e, **SET**, **CLEAR**, or **CLEAR ALL** for each of the 8 trigger levels. Operation setting may be also specified through a choice list (shown in Figure 5-37) by pressing <F2>.

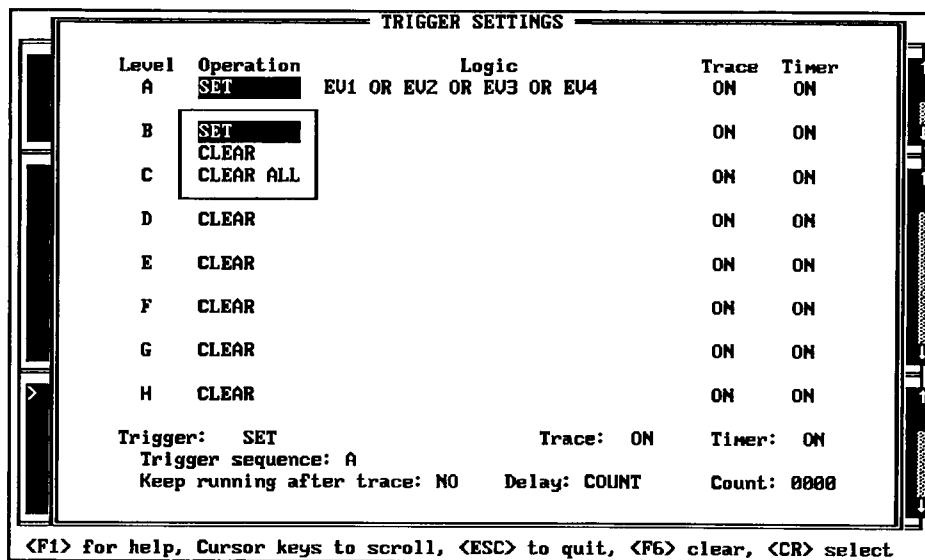


Figure 5-37 "Trigger Level Operation" Choice List

When **SET** operation is selected, the next three fields i.e., **Logic**, **Trace** and **Timer** will prompt user for further input. These fields are disabled when **CLEAR** and **CLEAR ALL** are selected. **CLEAR** operation clears the specified trigger setting only, while **CLEAR ALL** will clear all the eight trigger level settings whenever it is selected in any of the fields under **Level** operation.

The fields under **Logic** are for specifying trigger levels. Each level may be a single event or a logical construct of up to five Events with OR, AND and NOT. Only Events 1 to 5 can be used in the trigger level definitions. Events are automatically joined to the specified construct by a logical OR. Entries has to be typed in.

The fields under **Trace** are used to enable or disable trace program execution in the trace buffer. Use <space-bar> to toggle between "ON" (enable) and "OFF" (disable).

The fields under **Timer** are used to enable or disable time measurement of user's program execution.

Trigger Level...

MENU COMMAND

The trigger settings are set or cleared with the **Trigger** fields at the bottom half of the dialog box. Use <space-bar> to toggle between "SET" and "CLEAR". Selecting "CLEAR" will skip the subsequent fields. Selecting "SET" will prompt user to the subsequent fields under it, or as follows:

Trace: This field uses a <space-bar> "YES/ NO" toggle to enable or disable trace program execution in the trace buffer.

Timer: This field is used to enable or disable time measurement of user's program execution. Use <space-bar> to toggle between "ON" (enable) and "OFF" (disable).

Trigger sequence: This field is used to set trigger sequence of up to 8 trigger levels (identified as Levels A, B, C, D, E, F, G or H above) and mixing them with THEN... and IF... ELSE... connectives.

Keep running after trace: This field uses a <space-bar> "YES/ NO" toggle. Selecting "YES" will allow the emulation processor to free run after the trace ends. "NO" will stop the emulation processor one cycle beyond the location where the trigger condition was matched.

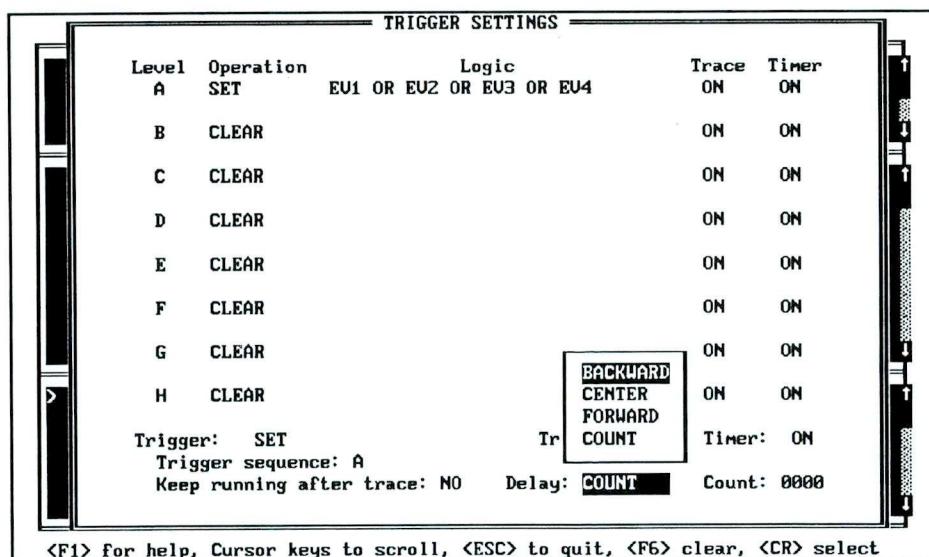


Figure 5-38 "Delay Count" Choice List

Delay: This field uses <F2> to display a delay count choice list as shown in Figure 5-38 above. Select desired delay setting, where:

FOrward is the delay count = 1FFFH or 7FFFH for 8K or 32K trace buffer respectively.

BAckward is the delay count = 0.

CEnter is the delay count = 0FFFH or 3FFFH for 8K or 32K trace buffer respectively.

count is a hexadecimal value (from 0H to 0FFFFH).

If "COUNT" is selected, user is prompt to the next **Count** field.

When emulation begins, the MICE immediately starts recording target system and emulation processor status in real-time. Data are recorded until cycle-count delay (Backward, Center, Forward or Count) and other trigger condition is matched.

Count: Specifies the number of cycle count (hex value from 0H to 0xFFFFH) in this field.

After all other trigger conditions have been matched, the trace will continue until the specified number of cycles has elapsed

Refer to MICE-III manuals for more details.

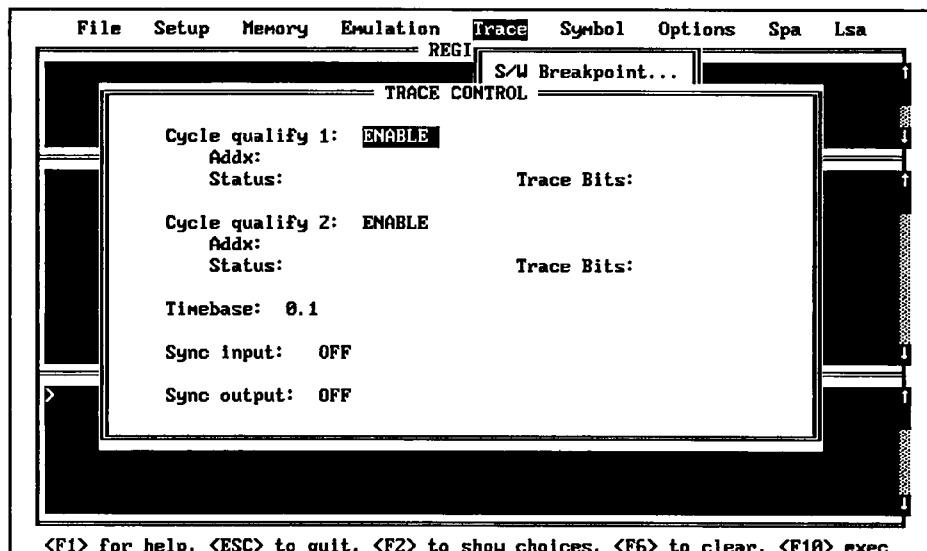
Trace Control...

MENU COMMAND

5.2.5.5 Trace Control Settings - Trace Control (for LAM-II only)

Equivalent command syntax:

refer to applicable MICE-III manual

**Figure 5-39 LAM-II "Trace Control" Dialog Box**

TRACE CONTROL dialog box provides MICE-III trace control settings for Cycle Qualify, Timebase and Synchronization commands in one display. One or all of the commands may be set and activated in one execution. This dialog box will pop-up displaying current MICE trace settings.

Cycle qualify field sets, enables, disables or clears cycle qualifier. Note that LAM-II provides two available qualifiers which perform "OR" function. Press **<F2>** to display the cycle qualify choice list as shown in Figure 5-40. The arguments specifies which machine cycles are to be recorded in the trace buffer. If "SET" cycle qualify is selected, user is prompt to the subsequent fields right under it or as follows:

MENU COMMAND

Trace Control...

Addx: Specify a hexadecimal address or range of addresses prefixed with "RAnge".

Status: Specify a processor status code in this field to indicate type of processor activity to be recorded.

Trace Bits: Specify a 1-byte hexadecimal setting for trace bits in this field.

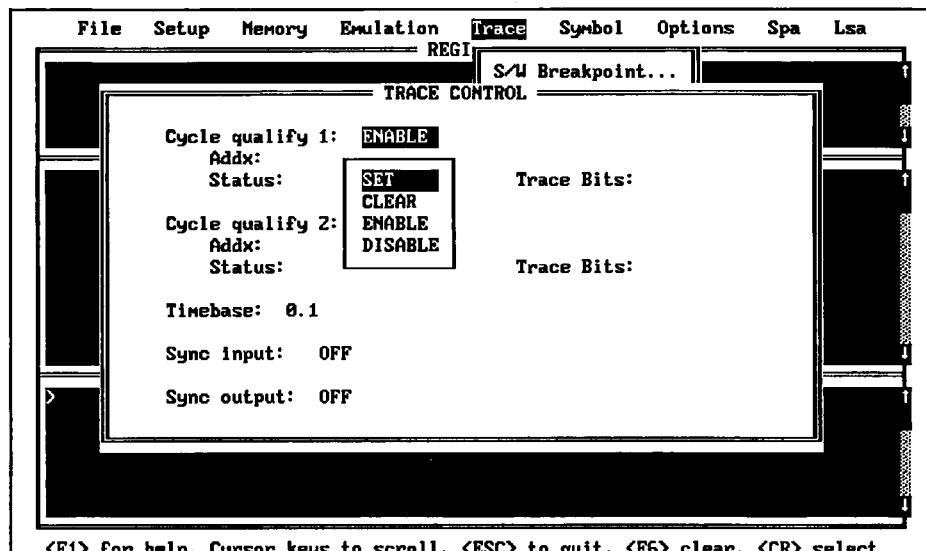


Figure 5-40 LAM-II 'Cycle Qualify' Choice List

Timebase will display a timebase (0.1, 1, 10, 100, 1000) choice list (Figure 5-41) after pressing <F2>. Select desired timebase. If none is selected, timebase will default at MICE setting. If <F6> is pressed to clear this field, no timebase command will be executed.

Trace Control...

MENU COMMAND

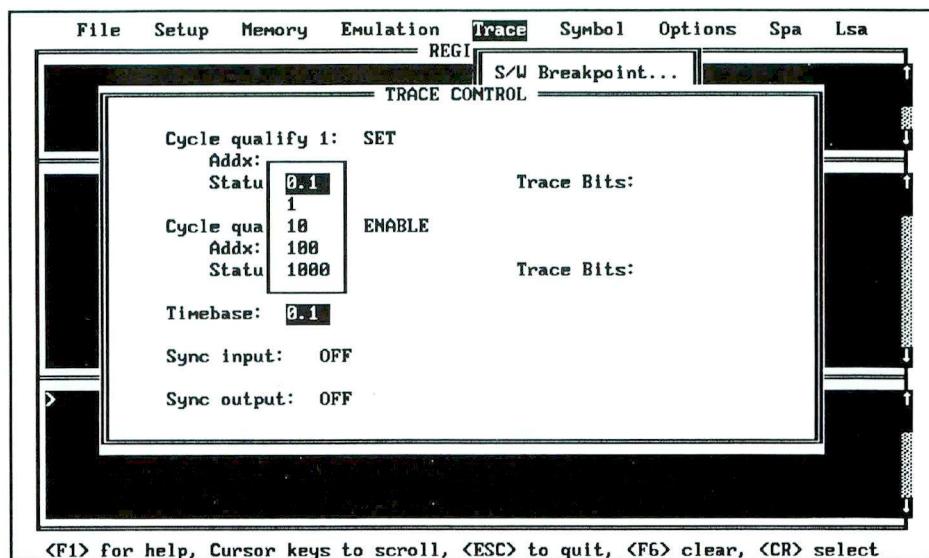


Figure 5-41 LAM-II "Timebase" Choice List

Sync input and **Sync output** fields set and clear synchronization. These fields will initially display current MICE setting. Use <space-bar> to toggle between "ON" and "OFF".

Refer to MICE-III manuals for more details.

MENU COMMAND

Trace Control...

5.2.5.6 Trace Control Settings - Trace Control

Equivalent command syntax:

refer to applicable MICE-III manual

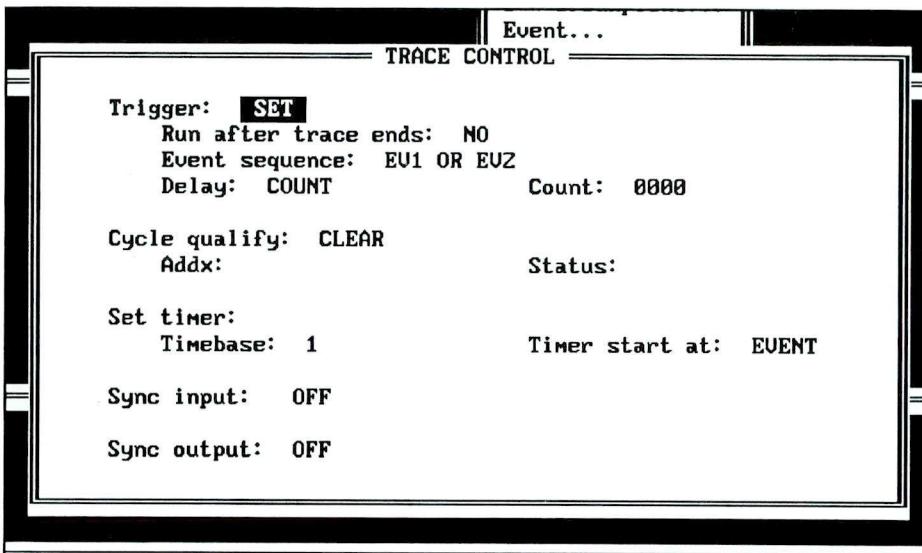


Figure 5-42 "Trace Control" Dialog Box

TRACE CONTROL dialog box provides MICE-III trace control setting for Trigger, Cycle Qualify, Timebase and Synchronization, in one display. One or all of the commands may be set and activated in one execution. This dialog box will pop-up displaying current MICE trace settings.

Trigger field is used to clear and set new trigger settings. Use <space-bar> to toggle between "SET" and "CLEAR". Selecting "CLEAR" will skip the subsequent trigger related fields right below it. Selecting "SET" will prompt user to the subsequent fields or as follows:

Trace Control...

MENU COMMAND

Run after trace ends: This field uses a <space-bar> "YES/ NO" toggle. Selecting "YES" will allow the emulation processor to free run after the trace ends. "NO" will stop the emulation processor one cycle beyond the location where the trigger condition was matched.

Event sequence: This field will display current MICE trigger setting (Figure 5-43). The trigger condition may be a single event or a logical construct of up to three breakpoints with And, Or and Then connectives. Only Events 1 to 3 can be used in the trigger definition. Note that no event may be specified more than once in the trigger setting. Press <F2> to display a choice list (figure below) showing all the event setting combinations.

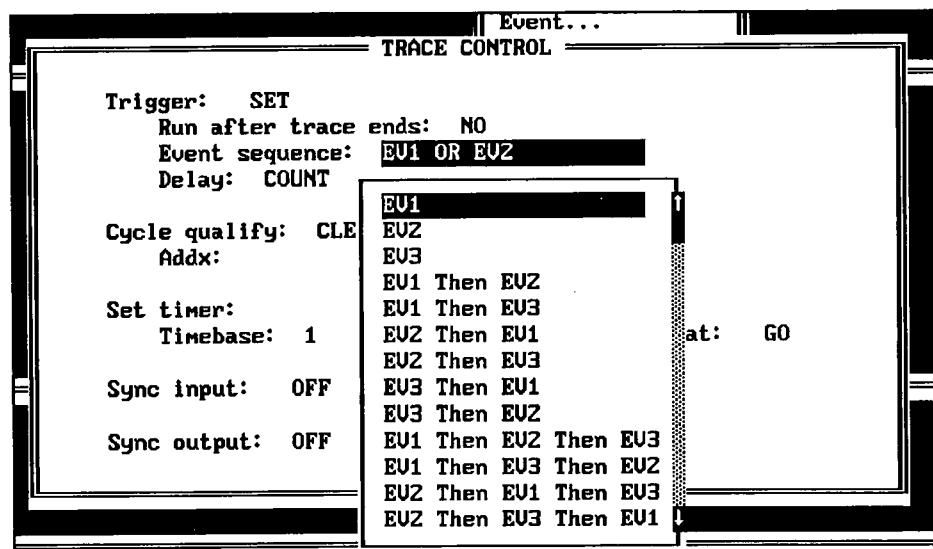


Figure 5-43 "Event Setting" Choice List

Delay: This field will display a cycle-count delay selections (Back-ward, Center, Forward and Count) as shown in Figure 5-44 below. Select desired delay setting. Selecting "COUNT" will prompt user to the next Count field.

When emulation begins, the MICE immediately starts recording target system and emulation processor status in real-time. Data are recorded until cycle-count delay (Backward, Center, Forward or Count) and other trigger condition is matched.

Count: Specifies the number of cycle count (hex value from 0H to 0FFFFH) in this field.

After all other trigger conditions have been matched, the trace will continue until the specified number of cycles has elapsed

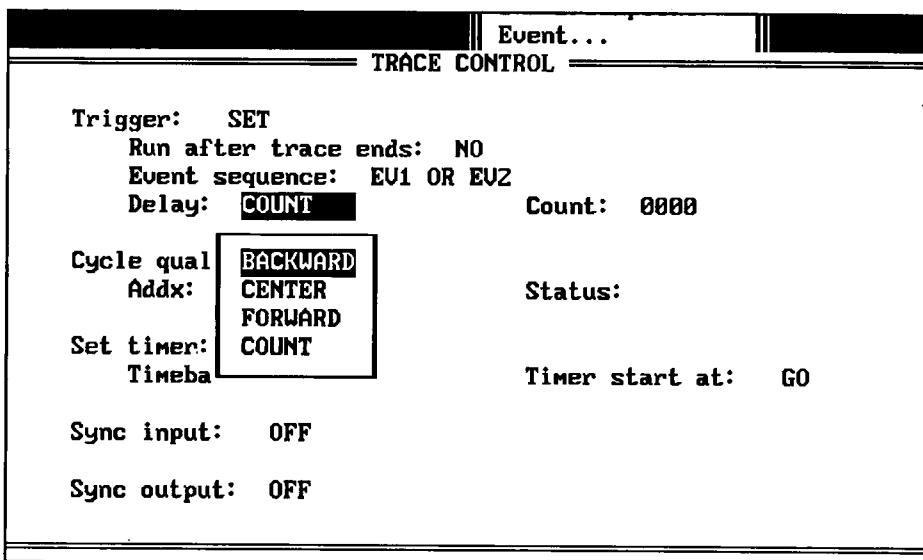


Figure 5-44 "Cycle-count Delay" Choice List

Cycle qualify field sets or clears cycle qualifier. Press <space-bar> to toggle between "SET" and "CLEAR". If "SET" cycle qualify is selected, user is prompt to the subsequent fields right below it or as follows:

Addx: Specify a cycle qualify hexadecimal address setting in this field.

Status: Specify a cycle qualify processor status code in this field.

Trace Control...

MENU COMMAND

The **Timebase:** and **Timer start at:** fields under the "Set timer" heading are used for timebase selection, where-

Timebase: will display a timebase (1, 10, 100, 1000, 10000) choice list (Figure 5-45) after pressing <F2>. Select desired timebase. If none is selected, timebase will default at MICE setting. If <F6> is pressed to clear this field, no timebase command will be executed.

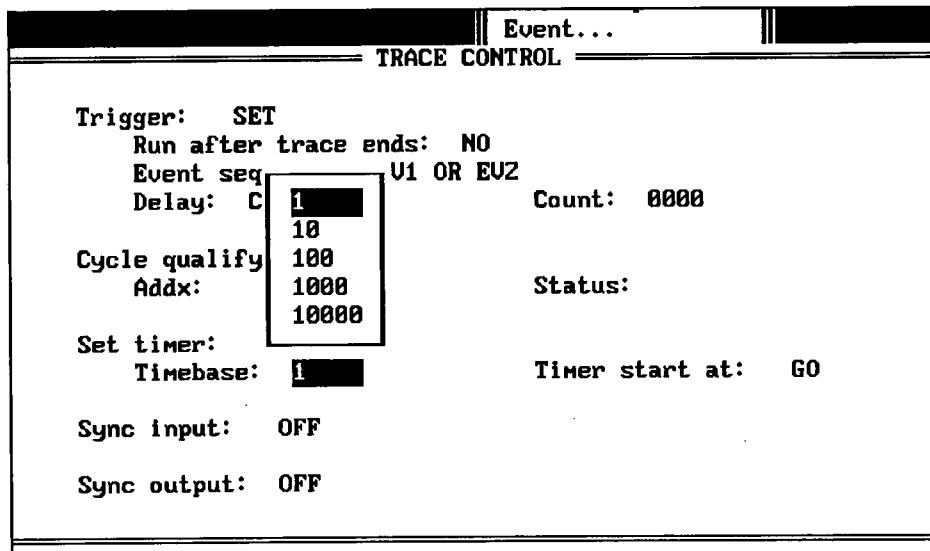


Figure 5-45 "Timebase" Choice List

Timer start at: Use <space-bar> to toggle and select between "GO" and "EVENT". Selecting "GO" will activate the timer when emulation starts. Selecting "EVENT" activates timer when the first event is encountered.

Sync input and **Sync output** fields set and clear synchronization. These fields will initially display current MICE setting. Use <space-bar> to toggle between "ON" and "OFF".

Refer to MICE-III manual for more details.

5.2.5.7 Trace with Backward Count - Backward Trace

Equivalent command syntax:

refer to applicable MICE-II manual

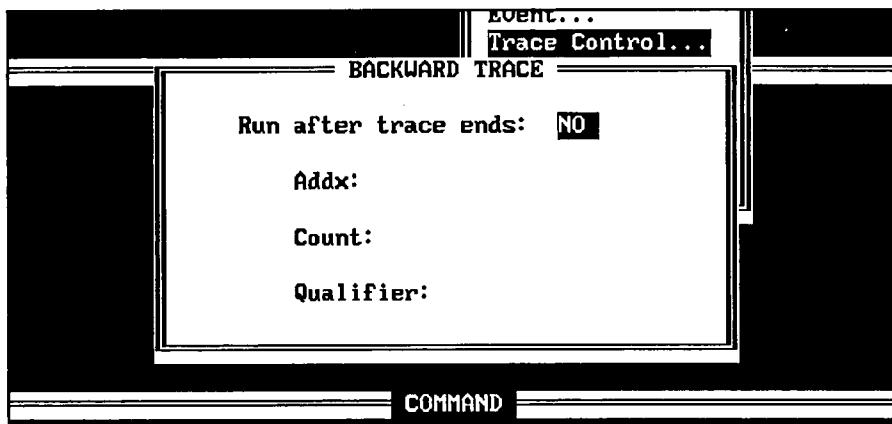


Figure 5-46 "Backward Trace" Dialog Box

Note that this command is applicable to MICE-II only.

BACKWARD TRACE dialog box starts real time emulation of the target and immediately begins recording target status until the specified trigger address is reached. Emulation stops when trace stops. Target CPU may however continue to run if the <space-bar> toggle under **Run after trace ends** is switched to "YES".

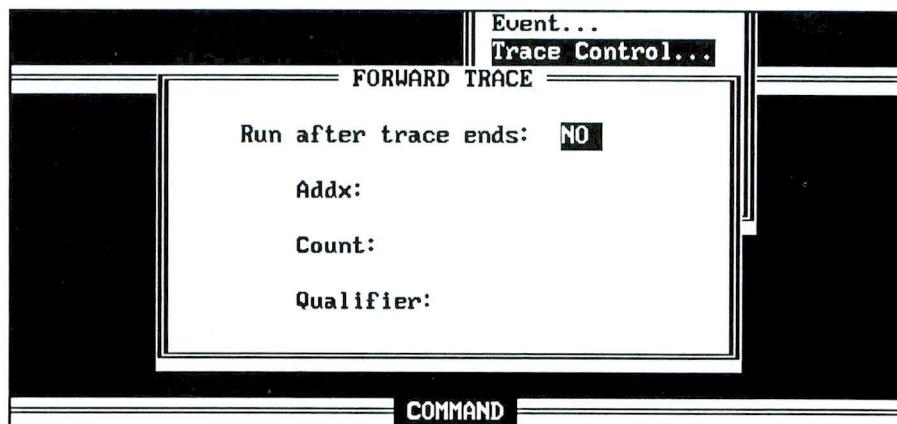
Backward trace starts real time emulation of the target and immediately begins recording target status until the trigger address specified in the **Addr:** field is reached.

Specify a "Backward count" (number of times the target condition has to be matched before trace stops) in the **Count:** field. A single qualifier may be also specified in the **Qualifier:** field to select the type of CPU activity associated with the trigger address. Note that "Backward Count" must be pre-specified before specifying a qualifier.

Refer to MICE-II manual for more details.

5.2.5.8 Trace with Forward Count - Forward Trace**Equivalent command syntax:**

refer to applicable MICE-II manual

**Figure 5-47 "Forward Trace" Dialog Box**

Note that this command is applicable to MICE-II only.

FORWARD TRACE dialog box starts real time emulation of the target and begins recording target status immediately after the specified trigger address is reached. Forward trace stops at the break address when the trace buffer is full or breakpoint is reached. Target CPU may however continue to run if the <space-bar> toggle under **Run after trace ends** is switched to "YES".

Forward trace starts real time emulation of the target and immediately begins recording target status until the trigger address specified in the **Addx** field is reached.

Specify a "Forward count" (number of times the target condition has to be matched before trace stops) in the **Count** field. A single qualifier may also be specified in the **Qualifier** field to choose the type of CPU activity associated with the trigger address. Note that "Forward Count" must be pre-specified before specifying a qualifier.

Refer to MICE-II manual for more details.

MENU COMMAND

List...

5.2.5.9 List Trace Buffer - List

Equivalent command syntax:

refer to applicable MICE manual

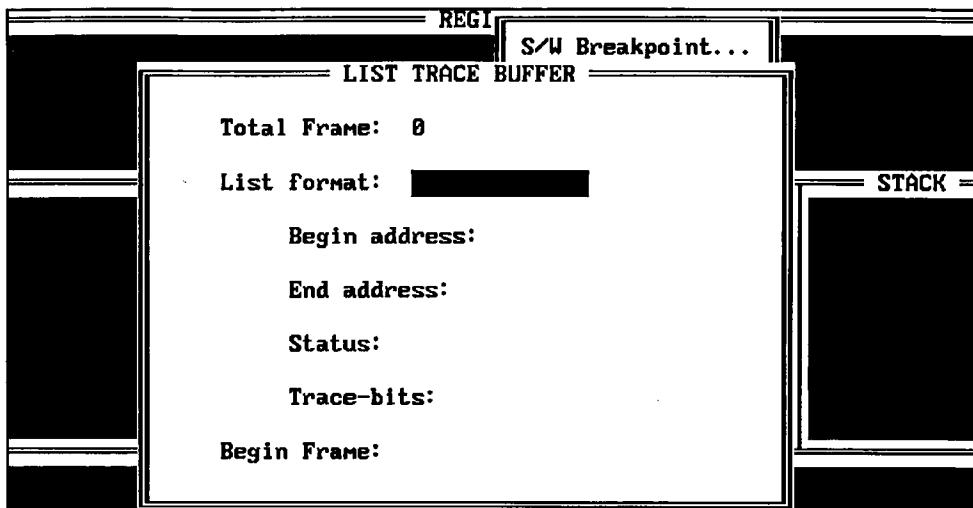


Figure 5-48 "List Trace Buffer" Dialog Box

LIST TRACE BUFFER dialog box lists information recorded in the trace buffer. The number of the last frame recorded in the trace buffer is displayed under the **Total Frame** field (first line) of this dialog box. Note that the frame count begins at zero, so the total number of frames is one more than the value on display. If "7FF" is displayed, the trace buffer is full.

With cursor moved to **List Format:** field, press <F2> to display choice list (Figure 5-49). Highlight the desired list format from this choice list. Selecting "Cycle record" from the choice list, will prompt user to the subsequent fields right below it or as follows:

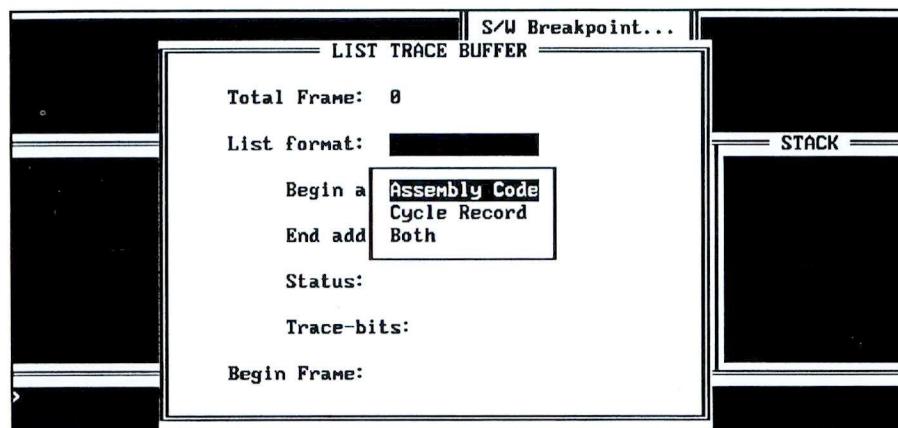


Figure 5-49 "List format" Choice List

Begin address: is the field where the starting address of the block for trace buffer listing is entered.

End address: is the field where the last address of the block for trace buffer listing is entered.

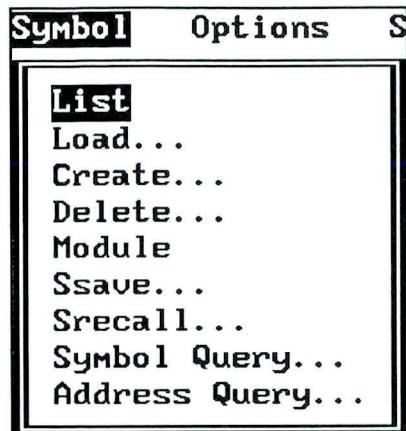
Status is the field to define the processor status codes representing specific type of processor activity. If none is entered, the default is for all cycles to be listed regardless of processor status.

Trace-bits is the field to assign trace bit level specifying that only cycles matching the indicated level are to be listed. If none is entered, the default is for all cycles to be listed regardless of trace bit levels.

The frame at which the listing is to begin is entered in the **Begin Frame** field. If none is entered, listing will begin at the last page of the buffer.

Refer to MICe manual for more details.

5.2.6 "Symbol" Pull-Down Menu Commands



When pulled down, the "Symbol" pull-down menu will provide commands for symbolic debugging information as illustrated in Figure 5-50. The following sections will explain these menu commands in further detail.

The "List" and "Module" commands provide no dialog boxes. They are directly executed from the pull-down menu.

"List" displays all symbols, including global symbols (from symbol file or created by "Create" command) and symbols in each module.

Refer to Section 4.1.21 for the equivalent "SYMBOL [All]" command syntax.

"Module" command displays all modules existing in the symbol file.

Refer to Section 4.1.16 for the equivalent "MODULE" command syntax.

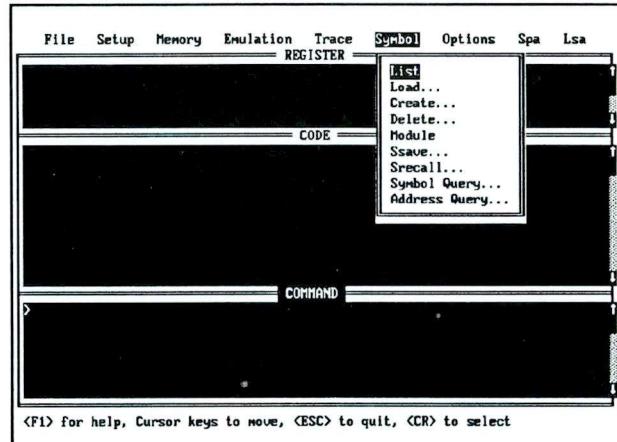


Figure 5-50 "Symbol" Pull-Down Menu

Load... MENU COMMAND

5.2.6.1 Load Symbol Table - Load

Equivalent command syntax:

LOAD filename

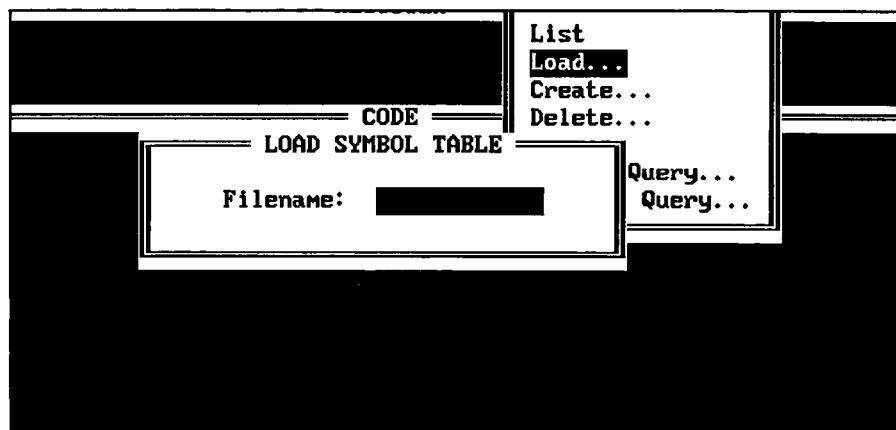


Figure 5-51 "Load Symbol Table" Dialog Box

LOAD SYMBOL TABLE dialog box loads symbol file from disk into USD-III. To load the symbol file enter symbol file to be loaded in the **Filename** field.

Refer to Section 4.1.14 for the equivalent "LOAD" command syntax.

MENU COMMAND

Create...

5.2.6.2 Create Symbol - Create

Equivalent command syntax:

CReate %symbol address

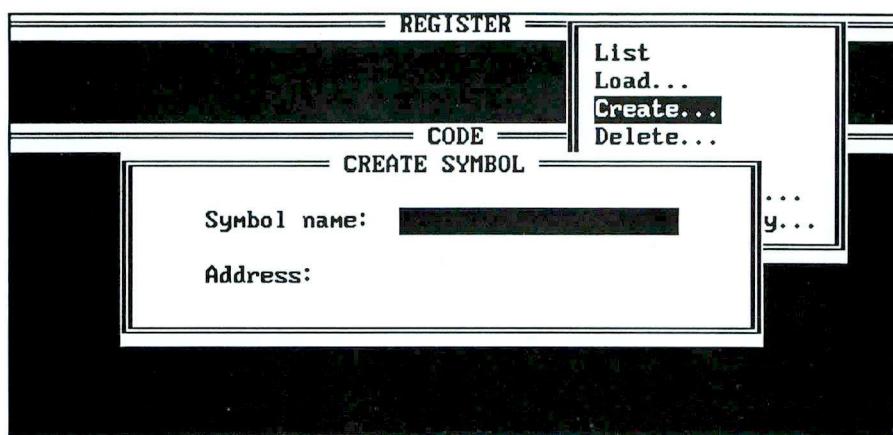


Figure 5-52 "Create Symbol" Dialog Box

CREATE SYMBOL dialog box is used to create a global symbol. Any symbol created by this command is temporary and effective only during the current debugging session. It will not be written back to the symbol file but may be saved as a disk file using the SAVE SYMBOL dialog box explained in Section 5.2.6.4. Note that the maximum number of symbols that can be save in a symbol table is 300.

Enter the symbol to be created in the **Symbol name** field. The symbol name may be of any alphanumeric characters.

NOTE

The percent sign <%> must be used as a prefix to every symbol name.

Specify the symbol address in hex value in the **Address** field. If the segment value is omitted from the symbol address and the symbol address is in segmented address format, the default CS value will be used as the segment value.

Refer to Section 4.1.3 for the equivalent "CReate" command syntax.

Delete... MENU COMMAND

5.2.6.3 Delete Symbol - Delete

Equivalent command syntax:

DELETE %symbol

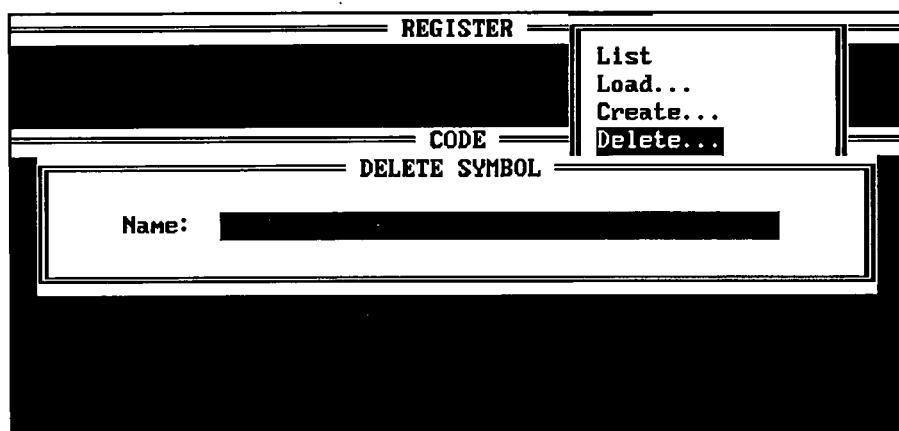


Figure 5-53 "Delete Symbol" Dialog Box

DELETE SYMBOL dialog box deletes symbols created by "Create" command. Specify the symbol to be deleted in the **Symbol name** field. Symbol should include the percent sign (%) prefix. Press <F2> to display listing of all the user created symbols.

Refer to Section 4.1.4 for the equivalent "DELet" command syntax.

MENU COMMAND

SSave...

5.2.6.4 Save Symbols to File - Ssave

Equivalent command syntax:

SSave filename

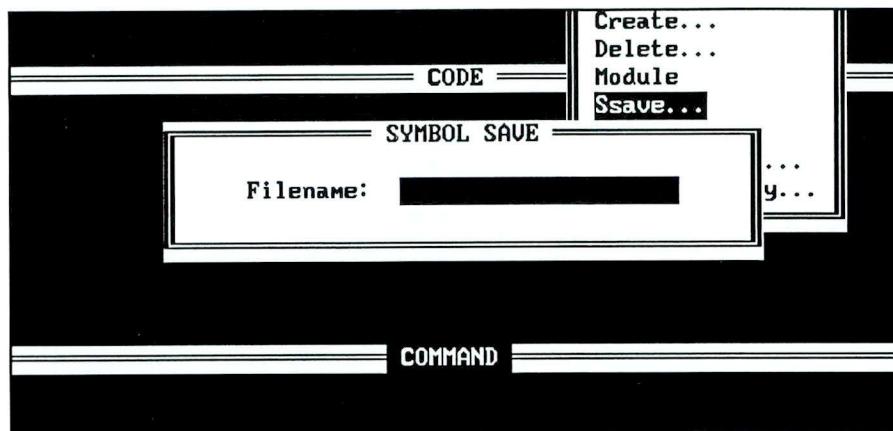


Figure 5-54 "Symbol Save" Dialog Box

SYMBOL SAVE dialog box saves symbols created by "Create" (Section 5.2.6.2) command. Specify the filename to save the symbols in the **Filename** field.

A maximum of 300 symbols can be saved from a created symbol table. The save symbols may be retrieved with **SRecall** command explained in Section 5.2.6.5.

Refer to Section 4.1.24 for the equivalent "SSave" command syntax.

Srecall...

MENU COMMAND

5.2.6.5 Recall Symbols from File - SRecall

Equivalent command syntax:

SRecall filename

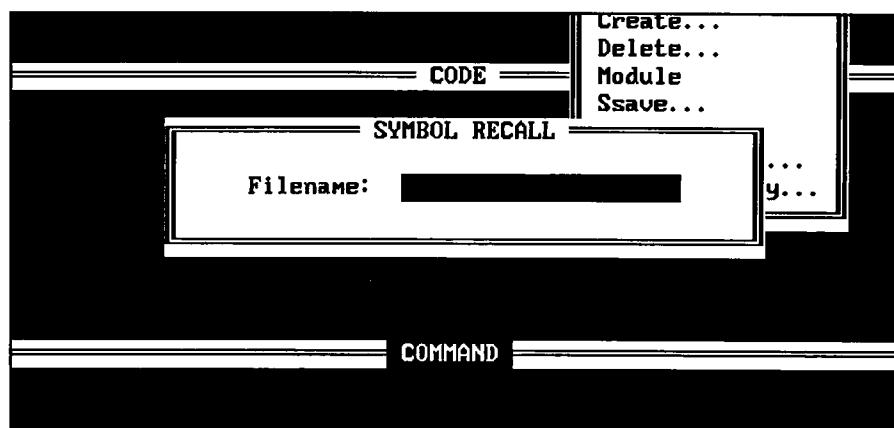


Figure 5-55 "Symbol Recall" Dialog Box

SYMBOL RECALL dialog box retrieves symbols saved by "Ssave" (Section 5.2.6.2) command. Specify the filename to retrieve the symbols from in the **Filename** field.

This command is useful when retrieving symbols to insert them into a newly created symbol table. If any symbols already exist in the new symbol table when the saved symbols are retrieved, the symbols from disk file are appended into the newly created symbol table. However, NOT all saved symbols may be retrieved into the new symbol table if the following conditions exists:

1. Identical symbol names exist.

Saved symbol names which already exist in the new symbol table can not be retrieved.

MENU COMMAND

Srecall...

-
2. The total number of symbols in the new symbol table exceeds 300.

The number of symbols that can be retrieved into the new symbol table is limited to its available symbol slots after new symbols has been created. Saved symbols are retrieved according to their filing sequence until the total 300 symbol limit is reached. The left-over are ignored.

Refer to Section 4.1.23 for the equivalent "SRecall" command syntax.

Symbol Query...

MENU COMMAND

5.2.6.6 Query for Symbol Name - Symbol Query

Equivalent command syntax:

```
QUEry [%module] %symbol
```

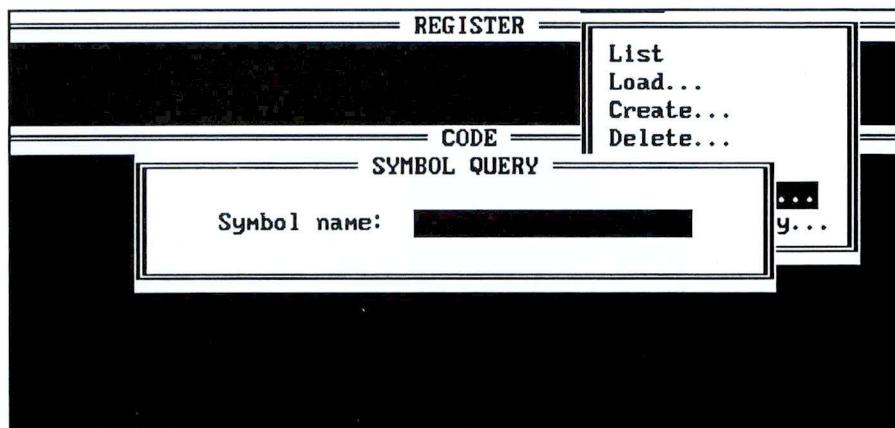


Figure 5-56 "Symbol Query" Dialog Box

SYMBOL QUERY dialog box is used to query for symbol name in the symbol table. Specify the symbol name to be queried in the **Symbol name** field. USD-III will search through the symbol table and list all occurrences of the specified symbol. Module name in "[%%module]%symbol" symbol format may also be specified, but only the specified module will be searched.

Refer to Section 4.1.18 for the equivalent "QUEry" command syntax.

MENU COMMAND

Address Query...

5.2.6.7 Query for Address - Address Query

Equivalent command syntax:

QUEry address

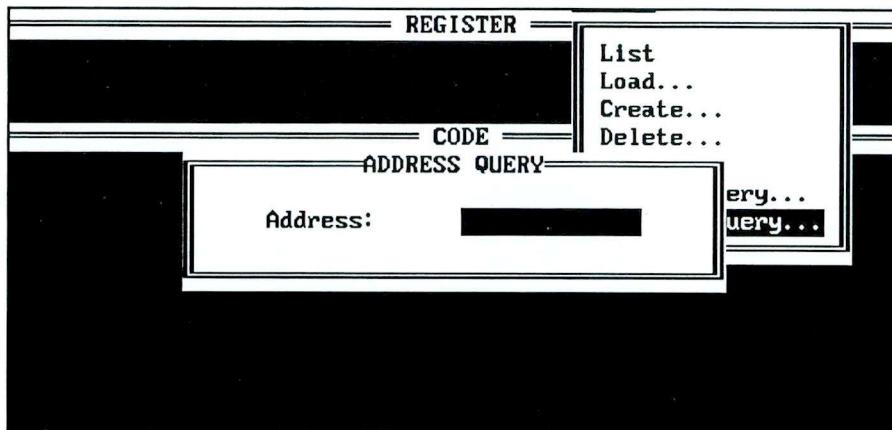


Figure 5-57 'Address Query' Dialog Box

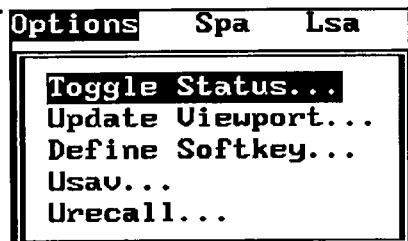
ADDRESS QUERY dialog box will query for symbol address in the symbol table. Specify the symbol address to be queried in the **address** field. USD-III will search through the symbol table and list all occurrences of the specified address.

Refer to Section 4.1.18 for the equivalent "QUEry" command syntax.

"Options" Pull-Down Menu

MENU COMMAND

5.2.7 "Options" Pull-Down Menu Commands



Options Menu (Figure 2-58) provides a check (✓) mark toggle to enable and disable Symbol/ Log/Journal/Software Breakpoint features as explained in the following sections. It also provides utility for saving and recalling window status and softkey definitions.

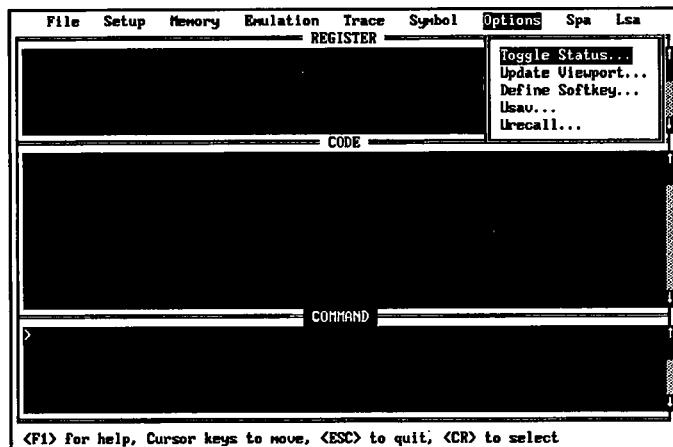


Figure 5-58 "Options" Pull-Down Menu

MENU COMMAND

Toggle Status...

5.2.7.1 Enable/Disable Symbol/Log/Journal/S/W Breakpoint - Toggle Status

Equivalent command syntax:

OPTIONS [{Symbol|Log|Journal|Breakpoint} {ON|OFF}]

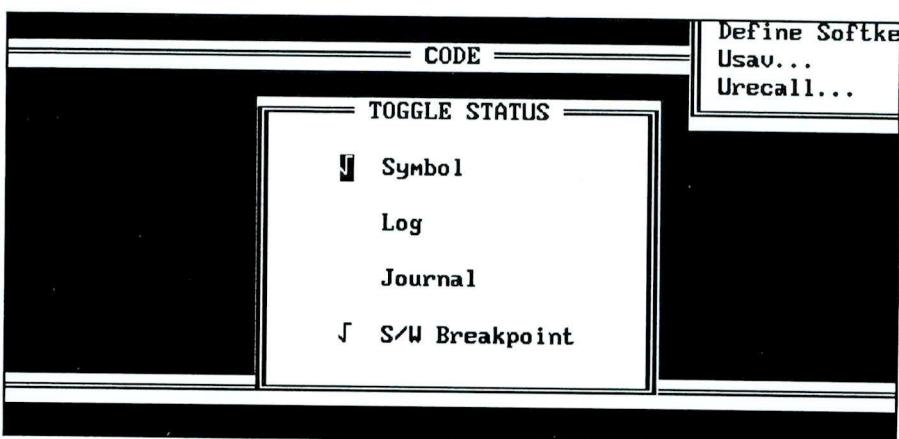


Figure 5-59 "Toggle Status" Dialog Box

To enable or disable Symbol, Log, Journal and S/W Breakpoint features, highlight desired feature and <space-bar> toggle. Notice the check mark (✓) turns ON and OFF at the cursor position. Features are enabled when check mark is ON. This command may be used for following applications:

Symbol Enable/ disables the search for symbolic output format for MICE Disassemble, single-Step and List trace buffer commands. Default status is "ON".

See "Symbol" pull-down menu (Section 5.2.6) for related commands.

Log Intermittently enable/ disables logging of executed command lines to the filename (as assigned in the LOG COMMAND FILE dialog box. See Section 5.2.1.4). The default is "ON" if a filename has previously been assigned, otherwise, it will default to "OFF".

Toggle Status...

MENU COMMAND

If no filename is assigned in the LOG COMMAND FILE dialog box, the following error message will display:

[USD ERROR]: NO LOG FILE!

Journal Intermittently enable/ disables logging of all COMMAND viewport activities (input and response) to the filename as assigned in the "LOG OUTPUT FILE" dialog box (Section 5.2.1.5). The default is "ON" if a filename has previously been assigned, otherwise, it will default at "OFF".

If no filename is assigned in the "Journal file" dialog box, the following error message will display:

[USD ERROR]: NO JOURNAL FILE!

S/W Breakpoint is used to continue (OFF) or stop (ON) program execution each time the software breakpoint setting set in the "SET SOFTWARE BREAKPOINT" dialog box (Section 5.2.5.1) is matched.

Refer to Section 4.1.19 for the equivalent "OPtions" command syntax.

MENU COMMAND

Update Viewport...

5.2.7.2 Enable/Disable Viewport Update - Update Viewport

Equivalent command syntax:

```
UPDATE {Breakpoint|CODE|Register|Stack|Trace|Data} {ON|OFF}
```

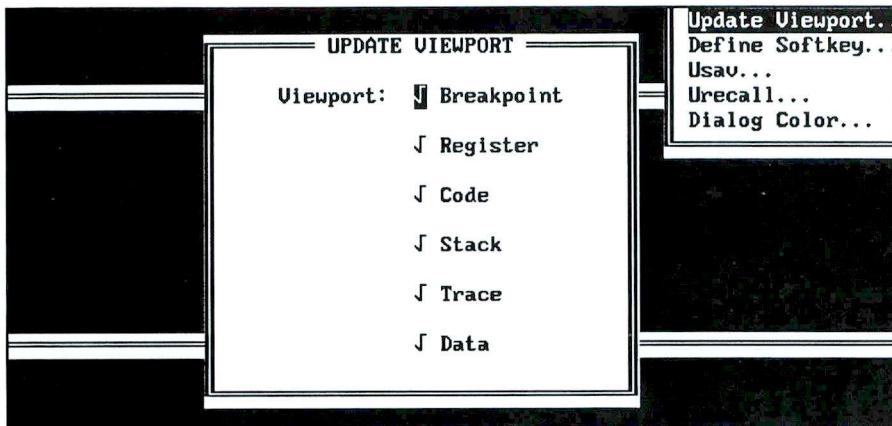


Figure 5-60 "Update Viewport" Dialog Box

UPDATE VIEWPORT dialog box shows the ON/OFF status of data updates feature on BREAKPOINT, CODE, REGISTER, STACK, TRACE and DATA viewports. A check mark (✓) prefix means the update feature is ON for the indicated viewport.

To enable/disable the update feature of a viewport, highlight the viewport and press <space-bar> toggle. Notice that the check mark (✓) turns ON and OFF at the cursor position.

NOTE

Disabling viewport update feature will speed-up USD-III commands execution.

Update Viewport...

MENU COMMAND

WARNING

If user wishes to switch into protected mode (available with MICE-II 80286, MICE-16 80286 and MICE-32 80386), viewport update feature must be disabled. Otherwise unexpected errors will result. For detailed description, please refer to the pertinent MICE manual.

Refer to Section 4.1.26 for the equivalent "UPDate" command syntax.

5.2.7.3 Define SoftKey Macro - Define Softkey

Equivalent command syntax:

```
SOFtkey {K1|K2|K3|K4|K5|K6|K7|K8|K9|K10} command_description
```

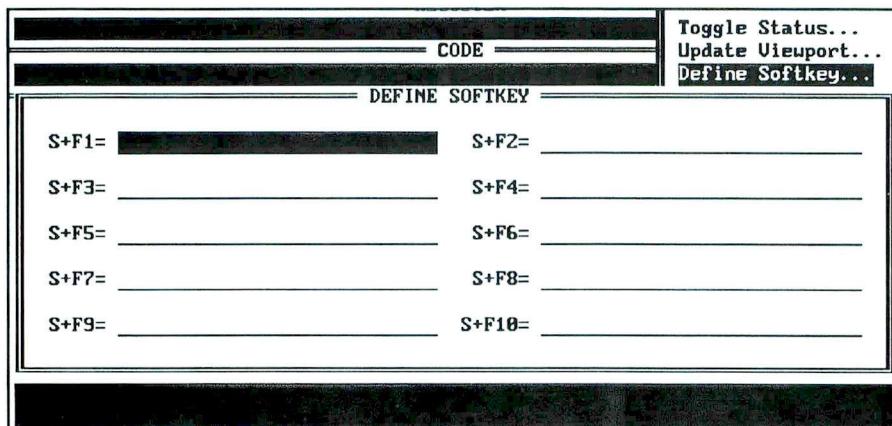


Figure 5-61 "Define Softkey" Dialog Box

DEFINE SOFTKEY dialog box is used to define function-keys <Shift-F1> to <Shift-F10> as the key macros to substitute one or more chained standard commands.

Using <↑><↓> or <Tab> key, access the desired function-key (K1...K10) field and key-in the command definition. The field could only display 30 characters. Excess characters are hidden. To move toward and view the hidden portion of the command definition, use <→><←> keys. <Home> and <End> may also be used to go to the beginning and end of the entry respectively. If more than one command are specified, they should be chained and separated with a semi-colon (;).

To redefine a key, edit the existing definition. To execute a defined key, press <Shift> and the function-key defined with the desired chained commands.

The key definitions are NOT automatically saved when quitting dialog box with <Esc>. They can only be saved either by pressing <F10>, or by pressing <Enter> repeatedly until cursor reaches the last key field and exits dialog box.

Refer to Section 4.1.22 the equivalent "SOFtkey" command syntax.

Usav...

MENU COMMAND

5.2.7.4 Save Current Window Status and Softkey Definitions - Usav

Equivalent command syntax:

USAve [W|K]



Figure 5-62 "Save Setting" Dialog Box

SAVE SETTING dialog box saves current window status (including screen layout, viewport color and viewport update flag, but NOT its contents) and softkey definitions into the USD3.SAV file.

When **SAVE SETTING** is accessed, a dialog box will pop-up to display a choice list as shown in Figure 5-62 above. <Space-bar> toggle the desired item and notice a check mark (✓) turns ON and OFF at the cursor position. Checked item will be saved when check mark is ON.

USD3.SAV only stores the most recent status saved by "Usav" command. It becomes the default window status and softkey definitions the next time USD-III is invoked.

To retrieve the saved window status and softkey definitions from USD3.SAV file while in a USD-III session, see "Urecall" menu command (Section 5.2.7.5).

Refer to Section 4.1.29 the equivalent "USAve" command syntax.

MENU COMMAND

Urecall...

5.2.7.5 Recall Window Status and Softkey Definitions - Urecall

Equivalent command syntax:

URECall [W|K]

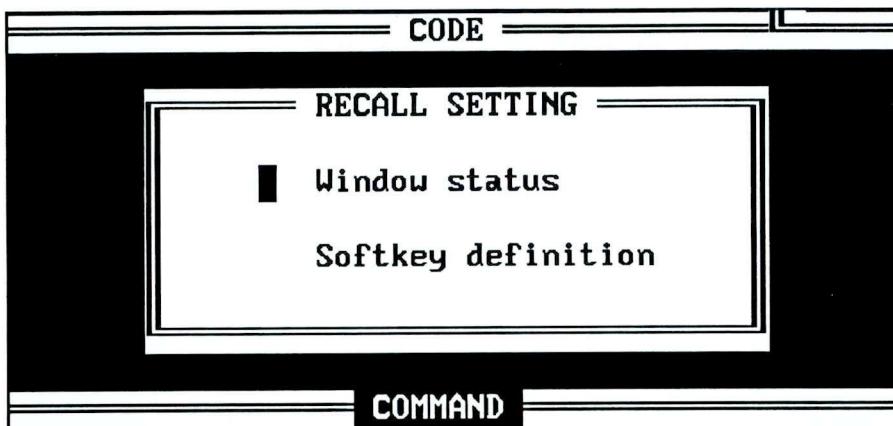


Figure 5-63 "Recall Setting" Dialog Box

RECALL SETTING dialog box retrieves window status (including screen layout, viewport/ dialog box color and viewport update flag, but NOT its contents) and softkey definitions stored in the USD3.SAV file.

When **RECALL SETTING** is accessed, a dialog box will pop-up to display a choice list as shown in Figure 5-63 above. Use <space-bar> to toggle the desired item and notice the check mark (✓) turns ON and OFF at the cursor position. Checked item will be saved when check mark is ON.

This command is useful when user wishes to change window status and/or softkey definitions while in a USD-III session.

Refer to Section 4.1.28 for the equivalent "URECall" command syntax.

5.2.7.6 Change Dialog Boxes Colors - Dialog Color

Equivalent command syntax:

none

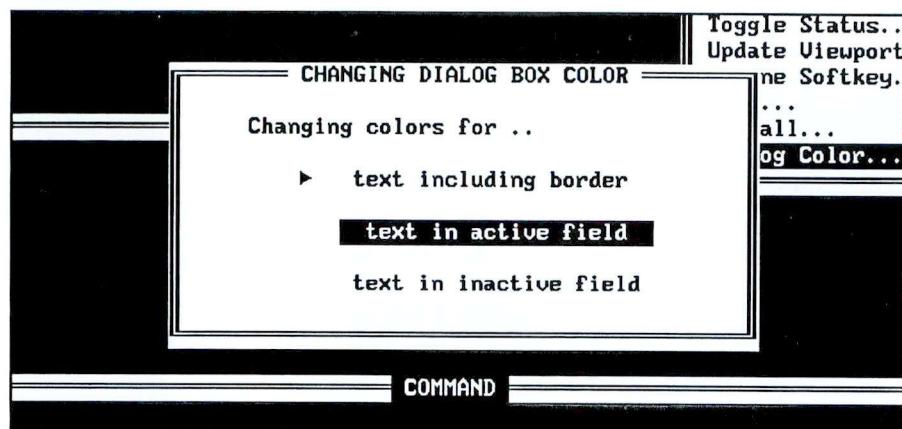


Figure 5-64 "Changing Colors" Dialog Box

CHANGING COLORS dialog box allows user to set and change colors of background, border, texts and fields in all USD-III dialog boxes. A total of 8 basic colors (black, blue, green, cyan, red, magenta, brown and white) plus the high intensity tint of each color are available for selection. To set or change colors of USD-III dialog boxes, follows the procedures below:

1. To change the color of the dialog box title and border, move pointer (arrow head) to the "**text including border**" field and press <F9>. Watch the title and border change or intensify its color each time <F9> is pressed. Then press <F10> to change color of the dialog box background. Again the background changes or intensifies its color each time <F10> is pressed. Keep on pressing <F9> and <F10> separately until the desired color appears.

-
2. To change the color of the active fields (the field where user places the cursor in position to input an argument or variable) in the dialog box, move pointer (arrow head) to the "**text in active field**" field and press <F9>. Watch the text on the field change or intensify its color and each time <F9> is pressed. Then press <F10> to change color of the field background. Again the field background changes or intensifies its color each time <F10> is pressed. Keep on pressing <F9> and <F10> separately until the desired color appears.
 3. To change the color of the inactive fields (other field in the dialog box other than the field where user places the cursor in position to input an argument or variable) in the dialog box, move pointer (arrow head) to the "**text in inactive field**" field and press <F9>. Watch the text on the field change or intensify its color each time <F9> is pressed. Then press <F10> to change color of the field background. Again the field background changes or intensifies its color each time <F10> is pressed. Keep on pressing <F9> and <F10> separately until the desired color appears.

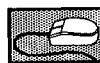
In case of monochrome display, <F9> will only result to either high intensity, blinking, blinking with high intensity, underline, reverse video or normal display. <F10> will only result to either normal display, underline or reverse video.

To cancel all new color settings and quit **CHANGING COLORS** dialog box, press <ESC>.

The new color setting takes effect immediately after the screen exit from **CHANGING COLORS** dialog box by pressing <Enter> repeatedly. However, unless otherwise saved into USD3.SAV file, the most recent color settings remains in effect only during the current USD-III session. To save the new color settings, use **Usav** command (Section 5.2.7.4) or answer "YES" to **Quit** command (Section 5.2.1.7) dialog box query "**Save screen and softkey ?**"

Dialog Color...

MENU COMMAND



Use the mouse with **CHANGING COLORS** dialog box as follows:

| move mouse cursor to- | click left-button to- | click right-button to- |
|---|--|--|
| any area in the dialog box other than the 3 fields. | save new setting to current USD-III session and exits from dialog box. | quit CHANGING COLORS dialog box and cancel all color changes. |
| the "arrow-head" pointed (active) field. | change foreground (text) colors. | change the background colors. |
| either fields other than the "arrow-head" pointed (active) field. | switch field to active. | (No operation.) |

5.2.8 "Spa" Pull-Down Menu Command

Equivalent line command syntax:

none



The "Spa" pull-down menu will provide Software Performance Analysis utilizing bar graphs to display result.

NOTE

1. SPA monitors the bus activity and address specification for consistency with the target bus width; i.e., for bus designs that access long-words, the address must be within long-word boundary. Bus designs that access words, the address must fall within word boundary.
2. Executing any SPA function will mask all breakpoint settings.
3. The emulation processor does not run 100% in realtime when executing any of these functions.
4. Pending completion of the undergoing MICE-III firmware updates, "Spa" is currently supported by MICE-II, MICE-16 H8/68K/80C186 and MICE-32 68020/68030 only.

Users will be advised accordingly (through MICE new version Release Notice) when the other MICE models are ready to support "Spa". The MICE Release Notice will also include field level instruction on how to change USD-III to enable "Spa" on the added MICE models.

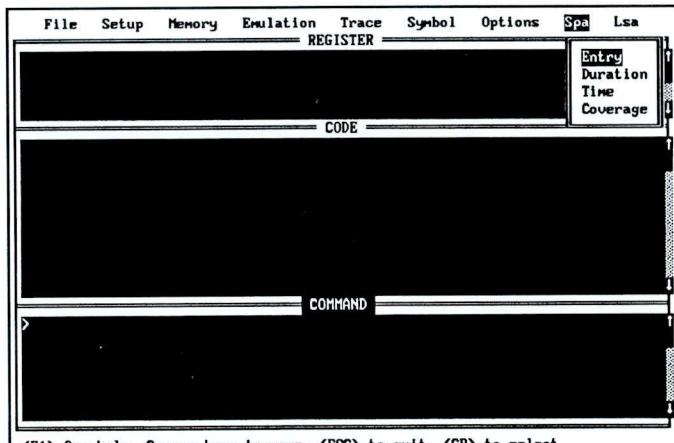


Figure 5-65 "Spa" Pull-Down Menu

"Spa" Pull-Down Menu

MENU COMMAND

SPA Applications

Software Performance Analysis can significantly assist in improving performance development of complicated software for realtime applications that requires minimum CPU execution time for each program module. This can be accomplished by accessing information in the program execution during realtime emulation of the target system.

Example: Supposing a user program has the following structure.

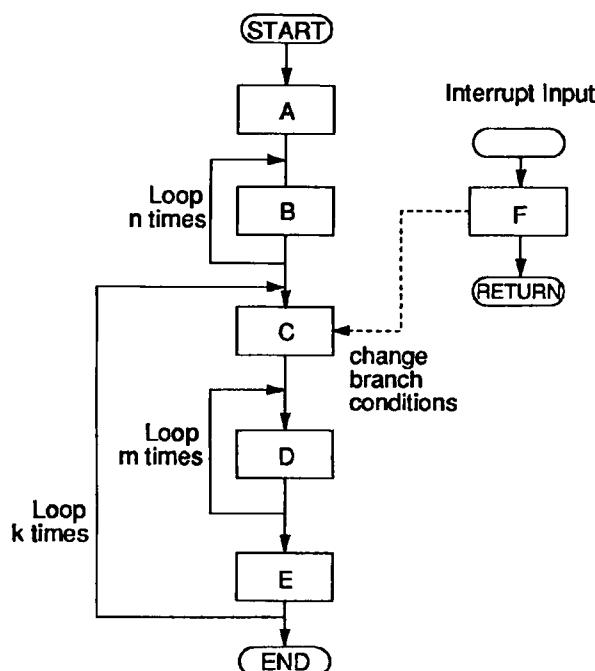


Figure 5-66 Structure of Sample Program

The sample program has two parts; the main program which includes 5 modules (A-E), and an interrupt service routine that controls the branch conditions in module C. If interrupts occur frequently, the branch conditions may differ each time module C is executed. To evaluate and improve the program performance with Software Performance Analysis, the following procedure was complied with:

1. Performed Module Duration Analysis (Section 5.2.8.2). The result as projected in Figure 5-68, was:

```
module A : 15%
module B : 12%
module C : 22%
module D : 30%
others     : 3%
```

2. The above information indicated that performance could be improved by rearranging the execution sequence, or as follows:

D → C → A → B → others.

3. However, the rearrangement could still be inefficient, because the number of times a module is executed (which is also a key performance factor) is not known. So the Entry Analysis (Section 5.2.8.1) was performed with "count" result shown in Figure 5-67 or as follows:

```
module A   1
module B  20
module C   8
module D  40
```

4. Now, with the information provided by Module Duration and Entry Analysis, a logical next approach is to rearrange the program modules as follows:

D → B → C → A.

5. Since the execution time for Module C depends on the branch conditions caused by interrupts, it is advisable to examine the time distribution pattern of program execution before modifying Module C. Therefore, Execution Time Analysis (Section 5.2.8.3) was performed with graphic result shown in Figure 5-69 and summarized below:

"Spa" Pull-Down Menu

MENU COMMAND

| | | |
|-------|-------|-----|
| | 10us | 0 |
| 10us | 100us | 0 |
| 100us | 1ms | 20% |
| 1ms | 10ms | 0 |
| 10ms | 100ms | 0 |
| 100ms | 1s | 80% |
| 1ms | 10s | 0 |
| 10s | 100s | 0 |
| 100s | | 0 |

6. Now check the conditional branch path to see if it is possible to make further improvements.

MENU COMMAND

Entry

5.2.8.1 Module Entry Analysis - Entry

Equivalent command syntax:

none

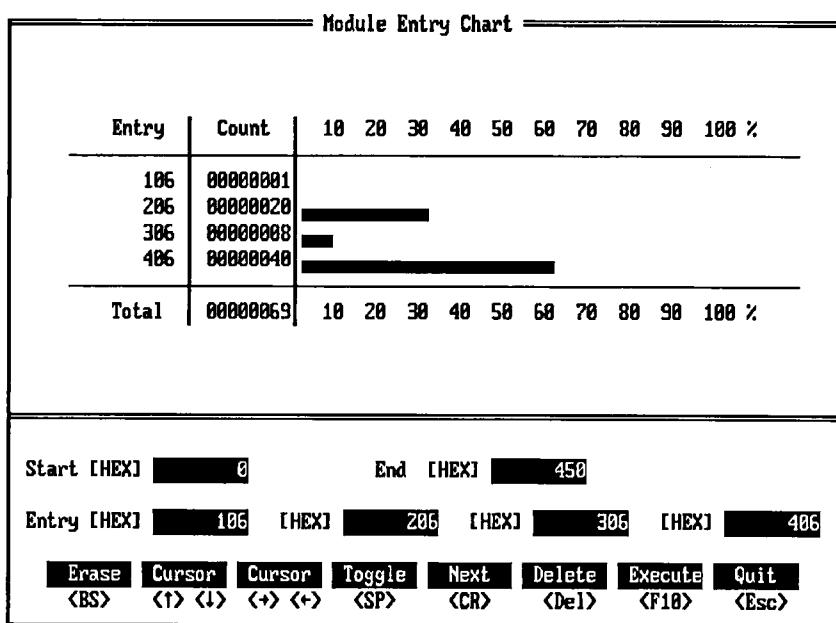


Figure 5-67 Module Entry Chart

"Entry" command will unveil the Module Entry Chart. The chart will display the total number of calls made to a group of specified program address. It also displays the count for calls made to each individual program address along with a bar graph indicating the percentage of processor activity involved for each address. The bar graph also permits quick comparison of processor activity for the specified address under different operating conditions.

The Module Entry Chart (Figure 5-67) has two sections, the "entry" (bottom) section and the "bar graph" (top) section. All data required for the performance analysis are input in the entry fields provided in the "entry" section, where-

- Start** is the "emulation start" address. It sets the PC to the location where emulation is to begin.
- End** is the "emulation end" address. It defines an optional end program address where emulation will stop.
- Entry** defines the address to be monitored. A maximum of 4 entries may be made per chart.

Use <space-bar> to toggle between [HEX] and [SYMBOL] entries. Press <space-bar> again to clear entries. Refer to the on-screen message line at the bottom of the "entry" section for other editing functions.

NOTE

DO NOT set "Start" address value identical with "Entry" address.

After all parameters are set, press <F10> to start performance analysis execution. The "bar graph" section will then start to fill with analytical results, where-

- Count** indicates the number of times program execution calls to each specified address.
- Bar graph** displays the percentage of processor activity for each specified program address.
- Total** indicates the overall amount of calls made to the specified address during emulation.

| MENU COMMAND | Duration |
|--------------|----------|
|--------------|----------|

5.2.8.2 Module Duration Analysis - Duration

Equivalent command syntax:

none

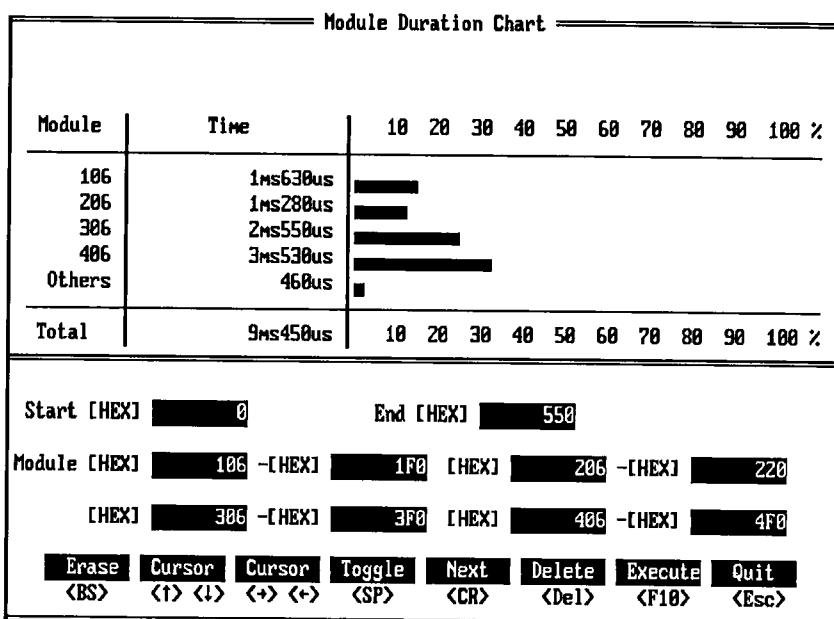


Figure 5-68 Module Duration Chart

"Duration" command will unveil the Module Duration Chart. The chart will display the overall duration of activity for a group (maximum of 4) of selected program modules. At the same time execution duration for each individual program module is also displayed along with a bar graph indicating percentage of processor activity involved for the module. Calculation begins each time execution enters the start address of a module and terminates only when the module end address is reached. This permits module activity to be graphed as a function of code performance, helping to evaluate the efficiency of the overall program or of specific modules.

Duration

MENU COMMAND

The Module Duration Chart also permits quick comparison of module activity for the specified address under different operating conditions. It can also reveal problems in a program, e.g., infinite loop or activity in forbidden areas, easy determination of the best/worst case condition of program options.

Note that "Duration" command is applicable to MICE-III only.

The Module Duration Chart (Figure 5-68) has two sections, the "entry" (bottom) section and the "bar graph" (top) section. All data required for the duration analysis are input in the entry fields provided in the "entry" section, where-

- Start** is the "emulation start" address. It sets the PC to the location where emulation is to begin.
- End** is the "emulation end" address. It defines an optional end program address where emulation will stop.
- Module** defines the "module range" start and end address of each program module to be monitored. A maximum of 4 modules may be defined in a chart.

Use <space-bar> to toggle between [HEX] and [SYMBOL] entries. Press <space-bar> again to clear entries. Refer to the on-screen "Help" message bar at the bottom of the "entry" section for other editing functions.

After all parameters are set, press <F10> to start module duration analysis execution. The "bar graph" section will then start to fill with analytical results, where-

- Time** indicates the overall duration of execution for each specified module during emulation.
- Bar graph** displays the percentage of processor activity for each specified program address.
- Total** indicates the overall length of execution time for the emulation processor. Note that shared execution time for nested modules is not included in this value.

MENU COMMAND

Time

5.2.8.3 Execution Time Analysis - Time

Equivalent line command syntax:

none

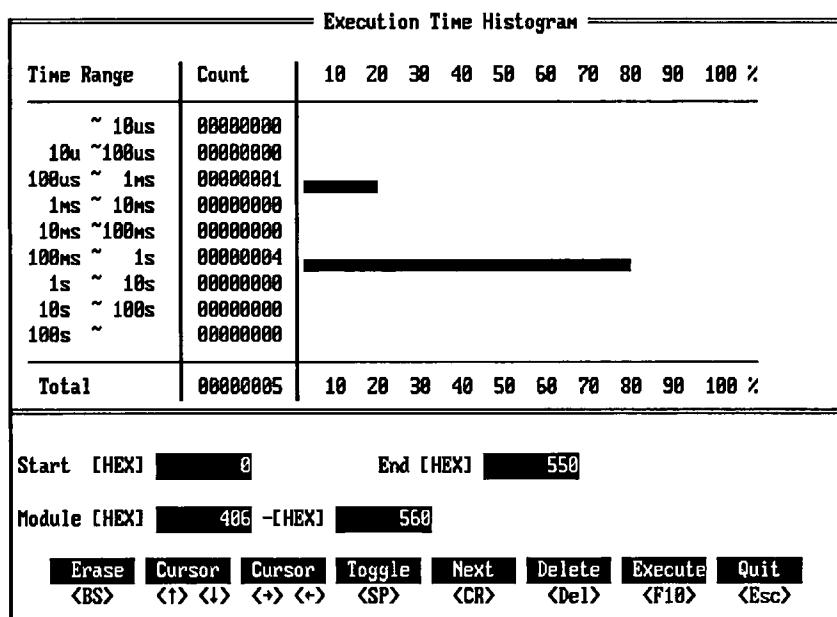


Figure 5-69 Execution Time Histogram

"Time" command will unveil an Execution Time Histogram. This histogram will display the activity distribution for a specified program module range. The frequency of execution within different time frames is displayed for a program module along with a bar graph indicating the percentage of processor activity involved for each specified time frame. This makes it easy to examine overall system activity or a specific program module for performance evaluation. Bottlenecks in program flow can be quickly identified, permitting dramatic improvement in execution time.

Time

MENU COMMAND

The Execution Time Histogram (Figure 5-69) has two sections, the "entry" (bottom) section and the "bar graph" (top) section. All data required for the time analysis are input in the entry fields provided in the "entry" section, where-

Start is the "emulation start" address. It sets the PC to the location where emulation is to begin.

End is the "emulation end" address. It defines an optional end program address where emulation will stop.

Module defines the "module range" start and end address of each program module to be recorded and analyzed.

Use <space-bar> to toggle between [HEX] and [SYMBOL] entries. Press <space-bar> again to clear entries. Refer to the on-screen "Help" message bar at the bottom of the "entry" section for other editing functions.

After all parameters are set, press <F10> to start module time analysis execution. The "bar graph" section of the chart will then start to fill with analytical results, where-

Time Range Defines the time range of execution interval for each recorded program module.

Count indicates the number of times the specified module is executed within different time ranges.

Bar graph displays the percentage of processor activity for each specified program address.

5.2.8.4 Code-Coverage Test - Coverage (for LAM-II only)

Equivalent line command syntax:

refer to applicable MICE-III manual

| CODE COVERAGE | | | |
|--|---------------------------------------|----------------------------|----------------------------|
| Bus status: | <input checked="" type="checkbox"/> I | <input type="checkbox"/> O | <input type="checkbox"/> S |
| Start addr: | End addr: | | |
| Operation | Range | Operation | Range |
| #01 <u>CLEAR</u> | _____ - _____ | #09 <u>CLEAR</u> | _____ - _____ |
| #02 <u>CLEAR</u> | _____ - _____ | #10 <u>CLEAR</u> | _____ - _____ |
| #03 <u>CLEAR</u> | _____ - _____ | #11 <u>CLEAR</u> | _____ - _____ |
| #04 <u>CLEAR</u> | _____ - _____ | #12 <u>CLEAR</u> | _____ - _____ |
| #05 <u>CLEAR</u> | _____ - _____ | #13 <u>CLEAR</u> | _____ - _____ |
| #06 <u>CLEAR</u> | _____ - _____ | #14 <u>CLEAR</u> | _____ - _____ |
| #07 <u>CLEAR</u> | _____ - _____ | #15 <u>CLEAR</u> | _____ - _____ |
| #08 <u>CLEAR</u> | _____ - _____ | #16 <u>CLEAR</u> | _____ - _____ |
| <↑> AltField <F7> ClearAll <F8> Reset <F9> List <F10> Execute <ESC> Quit | | | |

Figure 5-70 "Code Coverage" Dialog Box (80C186)

"CODE COVERAGE" dialog box test-checks the execution efficiency of a users program and displays the total rate of accessed memory ranges (actual accessed ranges / user specified ranges). The address ranges actually covered or accessed are also displayed.

Code coverage test is effective only in real-time emulation. It will not work under Cycle Step or Instruction Step commands. Note that the prefetched cycle during free-running is also in effect when performing code coverage test.

The dialog box pops up with the current MICE settings displayed on each field. To set, clear or change setting on each field, follow the steps below:

Coverage

MENU COMMAND

| CODE COVERAGE | | | |
|---------------|----------------------------|----------------------------|----------------------------|
| Bus status: | <input type="checkbox"/> I | <input type="checkbox"/> O | <input type="checkbox"/> S |
| Start addr: | <u>0000:0A00</u> | End addr: | <u>0000:B400</u> |
| Operation | Range | Operation | Range |
| #01 SET | <u>0000:0400-0000:0A40</u> | #09 CLEAR | _____ - _____ |
| #02 SET | <u>0000:0B00-0000:0B50</u> | #10 CLEAR | _____ - _____ |
| #03 SET | <u>0000:0C00-0000:0D00</u> | #11 CLEAR | _____ - _____ |
| #04 SET | <u>0000:3F00-0000:4A00</u> | #12 CLEAR | _____ - _____ |
| #05 SET | <u>0000:5000-0000:6500</u> | #13 CLEAR | _____ - _____ |
| #06 CLEAR | _____ - _____ | #14 CLEAR | _____ - _____ |
| #07 CLEAR | _____ - _____ | #15 CLEAR | _____ - _____ |
| #08 CLEAR | _____ - _____ | #16 CLEAR | _____ - _____ |

<↑> AltField <F7> ClearAll <F8> Reset <F9> List <F10> Execute <ESC> Quit

Figure 5-71 Setting Memory Block Range In "Range" Field (80C186)

Bus status Use <↑><↓> or <Enter> to move into the **Bus Status** field (this field becomes **Memory Types** for 68K/68020/68030) where bus status are displayed. Move cursor on the the bus status to be enabled and press <space-bar> to toggle a check mark (✓) prefix on it. Repeat the same routine to other bus status to be enabled, if any.

Start addr Move to **Start addr** field and input a hexadecimal address or a symbol specifying the starting point in the emulation CPU's program memory where emulation is to begin. Default is the MICE program counter. Aside from <Backspace>, <F6> may also be used to erase erroneously entered address in this field.

End addr Move to **End addr** field and input a hexadecimal address or a symbol specifying the last point in the emulation CPU's program memory where emulation is to end. If none is provided, emulation will continue until <Esc> is pressed. Aside from <Backspace>, <F6> may also be used to erase erroneously entered address in this field.

Operation Move to the appropriate field under **Operation** and press <space-bar> to toggle and make selection between CLEAR and SET. When CLEAR is selected, existing memory blocks corresponding to the CLEAR fields are cleared after pressing <F10>.

Range If SET **Operation** is selected, move cursor to the appropriate field under **Range**. Enter the starting point of the memory block range (in hexadecimal address) to be covered in the code coverage test on the left side field and the last point of the memory block on the right column field as shown in Figure 5-71. Repeat procedure until all desired memory block ranges are set. Press <F10> to execute.

As instructed in the dialog box (Figure 5-71) Message Line, use <**↑**><**↓**> to scroll through the set memory block ranges. To clear all the settings, press <F7>. <F8> is used to reset or re-initialize the code-coverage test without clearing the address ranges setting. To list the percentage of accessed memory and the address ranges actually covered (out of the memory block ranges set in Figure 5-71) for coverage test, as well as the Memory Dump or Assembly Code of such address ranges, press <F9>. This will expose user to the following "<F9> List" displays:

| 2.91 % of memory accessed in program ranges | | | |
|---|-------|----------|-------------------|
| BLOCKS ACCESSED | LOC | OBJ | ASSEMBLY CODE |
| 00A00 - 00A00 | | | SOURCE CODE |
| 00B00 - 01005 | 00B2C | 8B09 | MOV CX,[BX+DI] |
| 01234 - 01234 | 00B2E | 8B09 | MOV CX,[BX+DI] |
| 024B3 - 024B4 | 00B30 | 8B09 | MOV CX,[BX+DI] |
| | 00B32 | 8B09 | MOV CX,[BX+DI] |
| | 00B34 | 3E8B0A | MOV CX,DS:[BP+SI] |
| | 00B37 | 3E8B0A | MOV CX,DS:[BP+SI] |
| | 00B3A | 3E8B0A | MOV CX,DS:[BP+SI] |
| | 00B3D | 3E8B0A | MOV CX,DS:[BP+SI] |
| | 00B40 | 3E8B0B | MOV CX,DS:[BP+DI] |
| | 00B43 | 3E8B0B | MOV CX,DS:[BP+DI] |
| | 00B46 | 3E8B0B | MOV CX,DS:[BP+DI] |
| | 00B49 | 3E8B0B | MOV CX,DS:[BP+DI] |
| | 00B4C | 8A870A00 | AL,[BX]000A |
| | 00B50 | 8A87B324 | AL,[BX]24B3 |
| | 00B54 | 8A4708 | AL,[BX]08 |
| | 00B57 | 8A873412 | AL,[BX]1234 |
| | 00B5B | 8A87AB00 | AL,[BX]00AB |
| | 00B5F | 8A873412 | AL,[BX]1234 |
| | 00B63 | 8A840A00 | AL,[SI]000A |

<PgUp> Page↑ <PgDn> Page↓ <SP> Mode <F7> Zoom <CR> Display <ESC> Quit

Figure 5-72a Screen Display of "<F9> List" with ASSEMBLY CODE on Display (80C186)

Coverage

MENU COMMAND

The initial display will be as shown in the above figure. The Assembly Code for the highlighted address range or "Block Accessed" is displayed together with the percentage of memory and address ranges actually covered. Toggle <space-bar> (indicated as "Mode" in the Message Line of Figure 5-72a) to display the Memory Dump of the highlighted address range as shown below.

| 2.91 % of memory accessed in program ranges | |
|---|---|
| BLOCKS ACCESSED | MEMORY DUMP |
| 00A00 - 00A00 | 00 02 04 06 08 0A 0C 0E |
| 00B00 - 01005 | 0000 0000 0000 0000 0000 0000 0000 0000 |
| 01234 - 01234 | 00B10 0000 0000 038B 8B3E 3E03 038B 0F8B |
| 024B3 - 024B4 | 00B20 0C8B 0D8B 088B 088B 088B 098B 098B |
| | 00B30 098B 098B 8B3E 3E0A 088B 8B3E 3E0A 0A8B |
| | 00B40 8B3E 3E0B 088B 8B3E 3E0B 088B 878A 000A |
| | 00B50 878A 24B3 478A 8A0B 3487 8A12 AB87 8A00 |
| | 00B60 3487 8A12 8A84 8A00 B384 8A24 8844 848A |
| | 00B70 1234 848A 000B 849A 1234 858A 000A 858A |
| | 00B80 24B3 458A 8A0B 3485 8A12 AB85 8A00 3485 |
| | 00B90 8A12 8A00 8A00 B380 8A24 0840 808A 1234 |
| | 00BA0 808A 00AB 808A 1234 808A 000A 808A 24B3 |
| | 00BB0 408A 8A00 3480 8A12 AB80 8A00 3480 8A12 |
| | 00BC0 8A00 8A00 B380 8A24 0840 808A 1234 808A |
| | 00BD0 00AB 808A 1234 808A 000A 808A 24B3 408A |
| | 00BE0 8A00 3480 8A12 AB80 8A00 3480 8A12 0A81 |
| | 00BF0 8A00 B381 8A24 8841 818A 1234 818A 00AB |
| | 00C00 0000 0000 0000 0000 0000 0000 0000 0000 |
| | 00C10 0000 0000 0000 038B 8B3E 3E03 038B 0F8B |
| | 00C20 0C8B 0D8B 088B 088B 088B 098B 098B 098B |

<PgUp> Page↑ <PgDn> Page↓ <SP> Mode <F7> Zoom <CR> Display <ESC> Quit

Figure 5-72b Screen Display of '<F9> List' with MEMORY DUMP on Display (80C186)

The "BLOCKS ACCESSED" box may be removed from display to increase display screen area of either Assembly Code or Memory Dump by pressing "<F7>" as illustrated in Figures 5-72c and 5-72d below.

MENU COMMAND

Coverage

| | | 2.91 % of memory accessed in program ranges | | | | | | | | | |
|-------|----------|---|---------------|--|--|--|--|--|--|--|--|
| | | ASSEMBLY CODE | | | | | | | | | |
| LOC | OBJ | SOURCE CODE | | | | | | | | | |
| 00B2C | 8809 | MOV | CX,[BX+DI] | | | | | | | | |
| 00B2E | 8809 | MOV | CX,[BX+DI] | | | | | | | | |
| 00B30 | 8809 | MOV | CX,[BX+DI] | | | | | | | | |
| 00B32 | 8809 | MOV | CX,[BX+DI] | | | | | | | | |
| 00B34 | 3E8B0A | MOV | CX,DS:[BP+SI] | | | | | | | | |
| 00B37 | 3E8B0A | MOV | CX,DS:[BP+SI] | | | | | | | | |
| 00B3A | 3E8B0A | MOV | CX,DS:[BP+SI] | | | | | | | | |
| 00B3D | 3E8B0A | MOV | CX,DS:[BP+SI] | | | | | | | | |
| 00B40 | 3E8B0B | MOV | CX,DS:[BP+DI] | | | | | | | | |
| 00B43 | 3E8B0B | MOV | CX,DS:[BP+DI] | | | | | | | | |
| 00B46 | 3E8B0B | MOV | CX,DS:[BP+DI] | | | | | | | | |
| 00B49 | 3E8B0B | MOV | CX,DS:[BP+DI] | | | | | | | | |
| 00B4C | 8A870A00 | MOV | AL,[BX]000A | | | | | | | | |
| 00B50 | 8A87B324 | MOV | AL,[BX]24B3 | | | | | | | | |
| 00B54 | 8A4708 | MOV | AL,[BX]08 | | | | | | | | |
| 00B57 | 8A873412 | MOV | AL,[BX]1234 | | | | | | | | |
| 00B5B | 8A87AB00 | MOV | AL,[BX]00AB | | | | | | | | |
| 00B5F | 8A873412 | MOV | AL,[BX]1234 | | | | | | | | |
| 00B63 | 8A840A00 | MOV | AL,[SI]000A | | | | | | | | |

<PgUp> Page↑ <PgDn> Page↓ <SP> Mode <F7> Zoom <CR> Display <ESC> Quit

Figure 5-72c Screen Display of Zoomed ASSEMBLY CODE (80C186)

| 2.91 % of memory accessed in program ranges | | | | | | | | | | | |
|---|------|------|------|------|------|------|------|------|--------------------|--------------------|--------------------|
| MEMORY DUMP | | | | | | | | | | | |
| | 00 | 02 | 04 | 06 | 08 | 0A | 0C | 0E | | | ASCII-CODE |
| 00B00 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | | |>.... |
| 00B10 | 0000 | 0000 | 038B | 038B | 3E03 | 038B | 0F8B | |>.... | | |
| 00B20 | 0C8B | 0D8B | 088B | 088B | 088B | 098B | 098B | | | | |
| 00B30 | 098B | 098B | 883E | 3E0A | 0A8B | 883E | 3E0A | 0A8B |>....>... |>....>... |>....>... |
| 00B40 | 8B3E | 3E0B | 0B8B | 0B3E | 3E0B | 0B8B | 8784 | 000A | .>....>.... | .>....>.... | .>....>.... |
| 00B50 | 878A | 24B3 | 478A | 8A08 | 3487 | 8A12 | AB87 | 8A00 | ..\$.G...4.... | ..\$.G...4.... | ..\$.G...4.... |
| 00B60 | 3487 | 8A12 | 0A84 | 8A00 | B384 | 8A24 | 0844 | 8A8A | 4.....\$..D.. | 4.....\$..D.. | 4.....\$..D.. |
| 00B70 | 1234 | 848A | 00AB | 848A | 1234 | 858A | 000A | 858A | .4.....4.... | .4.....4.... | .4.....4.... |
| 00B80 | 24B3 | 458A | 8A08 | 3485 | 8A12 | AB85 | 8A08 | 3485 | \$..E...4.....4. | \$..E...4.....4. | \$..E...4.....4. |
| 00B90 | 8A12 | 0A80 | 8A00 | B380 | 8A24 | 0840 | 808A | 1234 |\$..0...4 |\$..0...4 |\$..0...4 |
| 00BA0 | 808A | 00AB | 888A | 1234 | 808A | 000A | 808A | 24B3 |4.....\$. |4.....\$. |4.....\$. |
| 00BB0 | 408A | 8A08 | 3480 | 8A12 | AB80 | 8A08 | 3480 | 8A12 | 0.....4.....4... | 0.....4.....4... | 0.....4.....4... |
| 00BC0 | 0A80 | 8A00 | B380 | 8A24 | 0840 | 888A | 1234 | 808A |\$..0...4.. |\$..0...4.. |\$..0...4.. |
| 00BD0 | 00AB | 808A | 1234 | 808A | 000A | 888A | 24B3 | 408A |4.....\$..0 |4.....\$..0 |4.....\$..0 |
| 00BE0 | 8A08 | 3488 | 8A12 | AB80 | 8A08 | 3480 | 8A12 | 0A81 | .4.....4.... | .4.....4.... | .4.....4.... |
| 00BF0 | 8A00 | B381 | 8A24 | 0841 | 818A | 1234 | 818A | 00AB |\$..A...4.... |\$..A...4.... |\$..A...4.... |
| 00C00 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | | | |
| 00C10 | 0000 | 0000 | 0000 | 038B | 038E | 3E03 | 038B | 0F8B |>.... |>.... |>.... |
| 00C20 | 0C8B | 0D8B | 088B | 088B | 088B | 098B | 098B | 098B | | | |

<PgUp> Page↑ <PgDn> Page↓ <SP> Mode <F7> Zoom <CR> Display <ESC> Quit

Figure 5-72d Screen Display of Zoomed MEMORY DUMP (80C186)

"Lsa" Menu Bar Command

MENU COMMAND

5.2.9 "Lsa" Menu Bar Command

Equivalent command syntax:

none

The "Lsa" menu bar menu will invoke the Logic State Analysis feature of USD-III. It initially displays the LSA Signal Specification Screen (Figure 5-73) where configuration for the LSA Signal Waveform is specified.

The resulting LSA provides a signal waveform display (Figure 5-74) for up to 16 channels. It also displays waveform diagram for a maximum of 88 trace points including up to 8 hardware trace points. All trace data recorded in the buffer can be conveniently displayed. Display of selected pin signals or hardware trace points, provides useful information on all bus, control line and status signal activity.

Prior to the waveform diagram display, a trace must be executed first to record relevant data into the trace buffer. Then specification of hardware signals and display channels for the waveform diagram can be made on the selection screen that follows. Easily defined parameters permit quick selection of display data for various debug requirements. Selection always display the most recent user input.

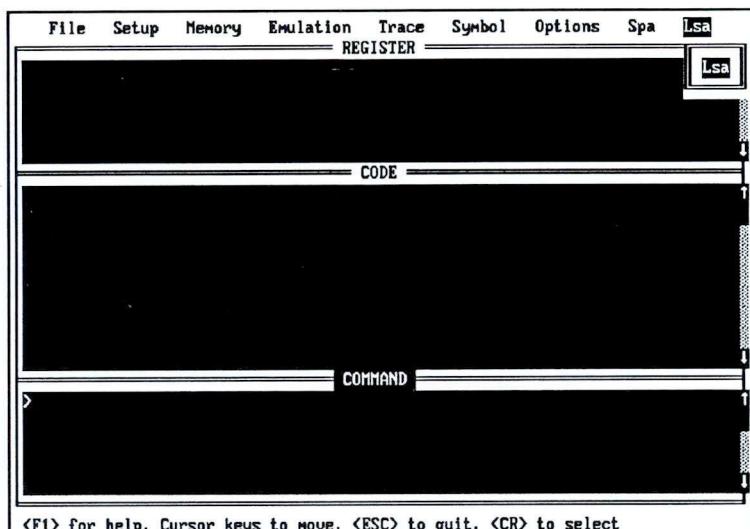
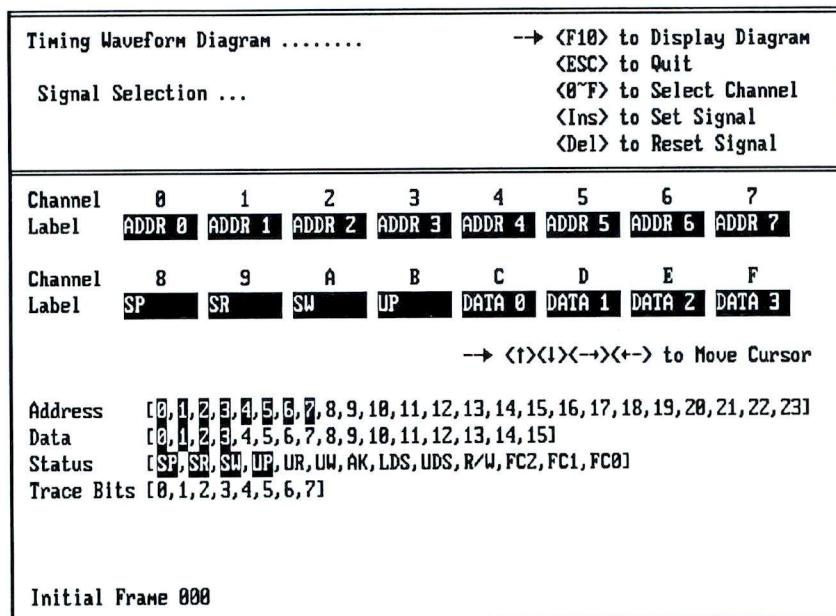


Figure 5-73 "Lsa" Menu Bar Menu

**Figure 5-74 Signal Specification for Logic State Analysis (68000)**

To configure a signal waveforms specification-

1. Use **<↑><↓><↔><→><←>** to select required signals by placing the cursor over the signals ("Address", "Data", "Status" and "Trace Bits" in the case of above figure) listed at the lower portion of Signal Specification screen (Figure 5-74). As each signal is on cursor, it will start to blink while waiting to be set into **Label** fields under "Channel" numbers (0 ~ F).
2. By default, pressing **<Ins>** each time a signal is selected, will automatically set the signal into **Label** fields in the consecutive order of "Channel" number. User may jump to other available "Channel" to set a blinking signal by pressing the number (0 ~ F) of the available "Channel".

In the above figure, "Address" signals were set by default. The "Data" signals were set starting at "Channel C" by pressing **<C>** followed by three **<Ins>**. Likewise, "Status" signals were set starting at "Channel 8" by pressing **"8"**.

"Lsa" Menu Bar Command

MENU COMMAND

A maximum of 16 signals may be selected in a waveform specification.

The resulting signal waveform will display from top to bottom in the same order as they are entered in the **Label** fields. See Figure 5-75.

3. If user wishes the waveform to start from frame other than "Frame 000" (the default), then move cursor to **Initial Frame** field (bottom line of the specification screen) to change the initial frame number.
4. To delete a signal, position cursor on the "Channel" number (0 ~ F) where signal is to be deleted, then press .
5. After all selections are set, press <F10> to display the waveform diagram (Figure 5-75).

Note that the most recent signal selection will remain effective in the current USD-III session only.

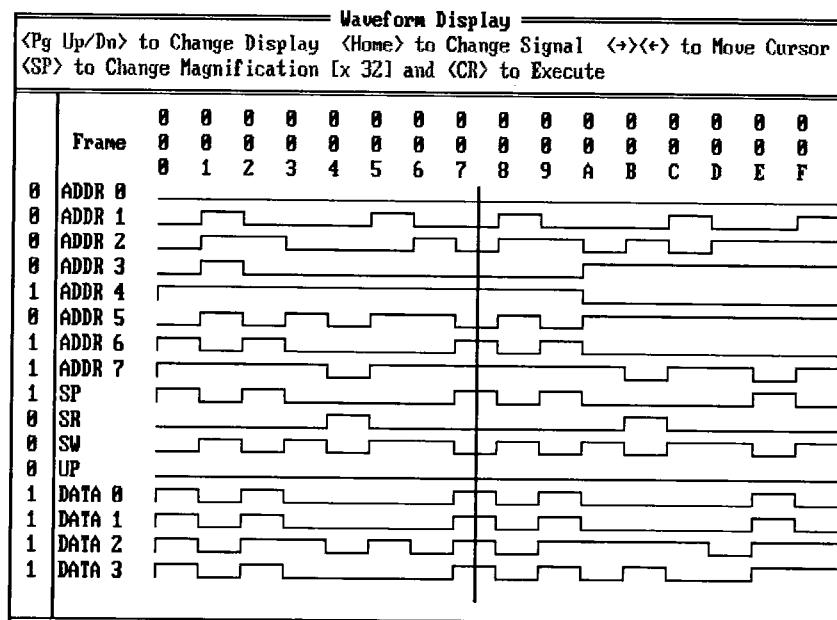


Figure 5-75 Logic State Analysis Waveform Display

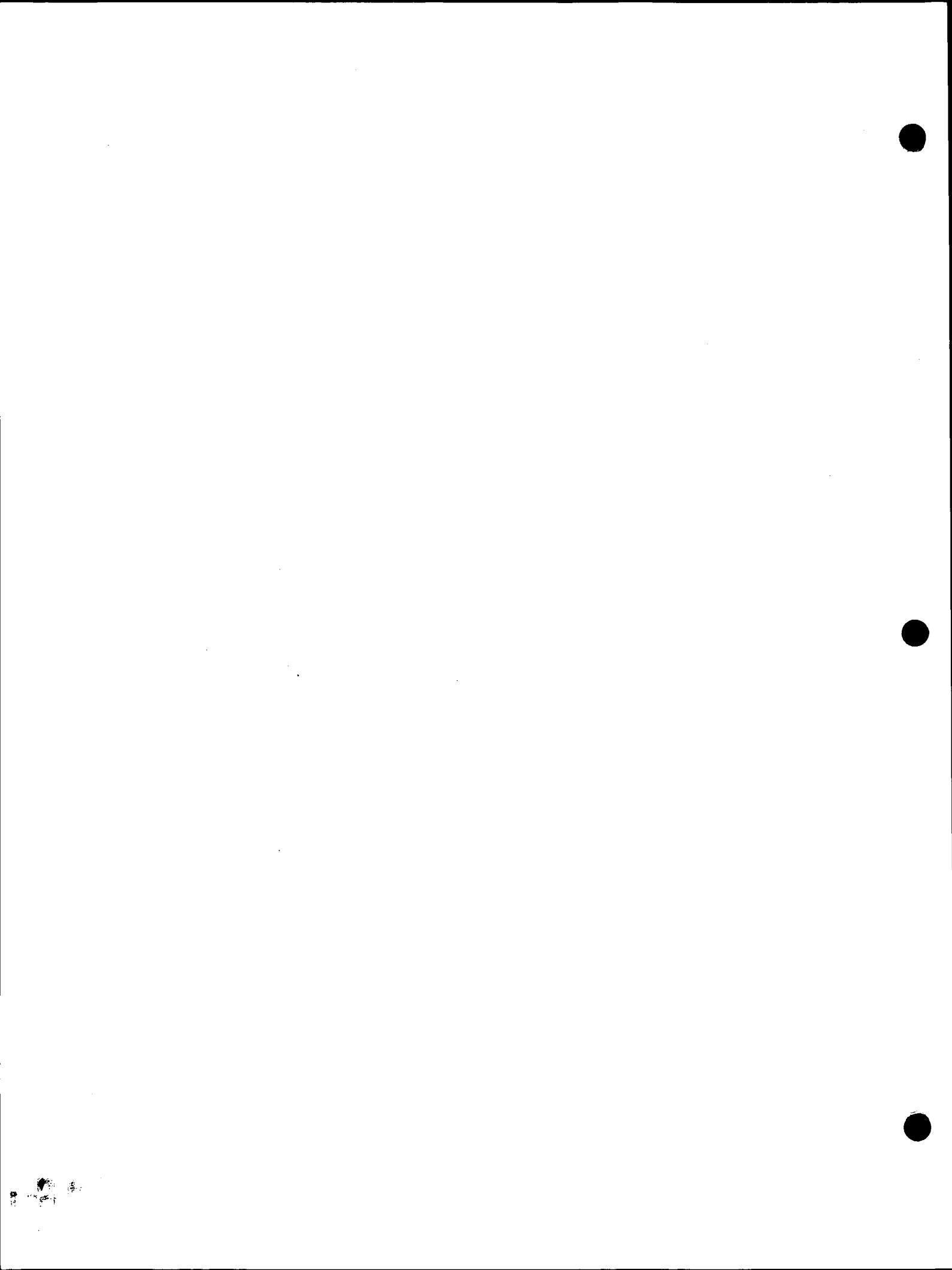
MENU COMMAND**"Lsa" Menu Bar Command**

The waveform screen display may be zoomed to 4/8/16/32/64 machine cycles frames per screen using the <space-bar>.

If the host computer is equipped with CGA/EGA/VGA, a vertical bar is provided to horizontally scroll the waveform one cycle frame at a time. This allows easy viewing the state of all signals in each frame. In the above figure, the vertical bar in the cycle frame number 8 position.

To display other signals currently recorded in the buffer, if any, press <PageUp> to scroll up or <PageDn> to scroll down one page at a time.

Press <Home> to return to the Signal Specification screen to specify a new set of signals.



Appendix A

CONVERTER PROGRAM

MICROTEK INTERNATIONAL INC.

A.1 INTRODUCTION

The converter CONV3.EXE is a universal symbol file conversion utility running under DOS environment. It extracts the symbol information from output files of the popular cross-assemblers or cross-compilers and converts them into a uniform symbol file known to USD-III. USD-III utilizes this two files to provide symbolic debugging information.

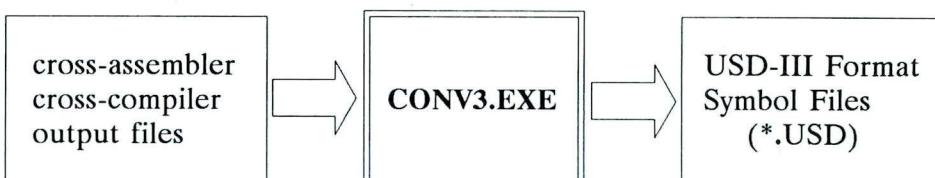


Figure A-1 CONV3.EXE Typical Operation

There is no limit on the number of symbols user may assigned in the symbol file. The symbol name or module name may contain from one to 40 characters. If more than 40 characters is specified, only the first 40 characters will be recognized. Note that all the symbol/module names are case-sensitive.

A.2 STARTING CONVERTER

To invoke the converter utility, enter the following syntax from DOS prompt:

>CONV3 inputfile [-O outputfile] [-R|-R3]

where:

- inputfile** is the exact input file (filename + file extension) to CONV3.
- O outputfile** is the option to specify a non-default output file without a file extension. If no output filename is specified, the converter will use the input filename, append ".USD" to it to form the output filename. If output filename is specified, the converter will automatically append ".USD" to it.

- R is option to be specified when converting 16/24-bit address length CPU in 2500AD format. See Section A.3.1 below for details.
- R3 is the option IAR/MPDS format using MICE-16 H8/500.

The output files will be created in the current working directory.

A.3 INPUT FILE FORMATS

The converter utility supports various input symbol formats from the following cross-compiler or cross-assembler vendors:

IAR Systems AB
S-75023 Uppsala, SWEDEN

Intel Corporation
3065 Bowers Ave., Sta. Clara, CA 95051 USA

Intermetrics, Inc.
733 Concord Ave., Cambridge, MA 02138 USA

2500AD Software, Inc.
109 Brookdale Ave., Buena Vista, CO 81211 USA

Microtec Research Inc.
2350 Mission College Blvd., Sta. Clara, CA 95054 USA

SYSROF, INSTAC, Japan Standard Association
25 Kouwa Bldg., #8-7 Sanbancho Chiyoda-Ku, Tokyo 102 JAPAN

A.3.1 2500AD Microtek Symbol Format

The converter utility takes the 2500AD's Microtek symbol format file as the input. Only the ".SYM" is required, which is the output of the 2500AD's Microtek symbol format file as the input.

To incorporate symbol information into the ".SYM" file with cross-assembler (versions 4.0 or higher required) the "SYMBOLS" directive should appear in the source file with the following linker options specified:

linker options: M

where:

M is to generate Microtek symbol format file.

The "-R" switch described in Section A.2 above has to be specified when converting 16/24-bit address length CPU in 2500AD symbol format file. Without the "-R" switch, the Motorola address convention is applied by default; i.e., the high and low bytes positions of the address appearing in the input file are kept in the same positions (not reversed). If the "-R" is specified, the Intel address convention will be implemented whereby CONV3 will force reverse the high and low bytes positions of the address to obtain the correct symbol address.

The general rule therefore is to apply Motorola address convention (without "-R") when Motorola CPUs, such as 6809, 68000, 68010 and 68008 are involved. Likewise, Intel address convention (with "-R") is to be applied when working with Z80, 64180, 6502 and 8051 CPUs.

A.3.2 IAR System Format¹

The converter utility only accepts the MPDS symbol format (same with 2500AD's Microtek symbol format) from IAR XLINK output version 3.12A or higher. Only ".SYM" is required.

To create ".SYM" file, the following options and parameters are required by XLINK:

- Fmpds
- Fmpds-i These three options can create MPDS symbol format files but
- Fmpds-m with different hex format.
- o filename* specifies the filename for ".SYM" file

**Specified filenames should be identical.*

¹ -R3 is only for IAR/H8 (MPDS format) compiler + MICE-16 H8.

Note that since MPDS symbol format is equivalent to 2500AD's Microtek symbol format, the "-R" option in the converter command line (discussed in Section A.3.1 above) is also applicable to IAR format. But for CPU H8/500, the -R3 switch should be given in the command line.

A.3.3 Intel OMF Format

The converter utility accepts Intel OMF51, OMF86 and bootloadable symbol formats for 8051, 8086, 80286 and 80386 respectively. The OMF files for these formats can be created with Intel iC86, C386, PL/M-86, 286 and other Intel tools. The table (Table A-2) below lists the different options required for various steps in creating the OMF file with Intel tools.

| FORMAT | Compile | Link/Bind | Locate/Build |
|--------------|---------|---------------------|--------------|
| OMF51 | / DEBUG | --- | --- |
| OMF86 | / DEBUG | --- | --- |
| Bootloadable | / DEBUG | / DEBUG / NOLOAD | --- |

Table A-2 Applicable Options for Intel Tools

A.3.4 Intermetrics Object Language Format

The converter utility requires ".AB" file as input from Intermetrics 68-series InterTools, but needs OMF input from 86-series InterTools.

To generate ".AB" file, apply the following options:

| | |
|-----------------|-------|
| compiler option | -d |
| link option | -x -o |
| locate option | -o |

Transfer the ".AB" file to OMF format through InterTools FORMATTER.

A.3.5 MRI Symbol Format

The converter utility accepts MRI (Microtec Research Inc.) formats for five different types of CPUs; 68000/020, 64180, Z80, 8051 and 8086. In addition to MRI's assembler and loader, MCC86, MCC180, MCC68K, MCCZ80 and ASM51 are also required to create their own respective absolute files.

For CPU 8086, both the ".ABS" and ".MAP" files are required. They should have the same name and reside in the same directory. But for CPU 68000/020, Z80, 64180 and 8051, only the ".ABS" file is required. The ".MAP" file is optional. There will be global symbols available if the ".MAP" file is given when converting for CPU Z80, 8051 and 68000/020. Both files are the output from MRI loader and should bear identical filenames and reside in the same directory. The following table lists the compile, assembly and load options required for each of the CPUs:

| CPU | Linker | Compile | Assembly | Load |
|----------|--------------|---------|-----------|---------|
| 6800/020 | LNK68K V6.6A | /g | /G/D/CASE | D |
| 64180 | LOD180 V2.0e | /DEBUG | /B/G | D,S,T,X |
| Z80 | LODZ80 V5.2 | /DEBUG | /B/G | D,S,T,X |
| 8051 | LOD51 V2.0J | --- | /DEBUG | D,S,T,X |
| 8086 | LOD86 V4.6I | /DEBUG | /DEBUG | D,S,T,X |

Table A-3 Applicable Options for MRI Tools

NOTE

*The converter utility also accepts Motorola S-Record format output file for 68000/020 created from the above option.
The MRI-IEEE can NOT be recognized by the converter.*

A.3.6 OM/LM Format (SYSROF - SYmbol-information, Standard Relocatable Object Format)

The converter utility accepts the ".ABS" file as the input. This is only for CPU H8. In order to include the symbol information into the ".ABS" file, the "/ DEBUG" switch should be given while compiling and linking the user programs.

Appendix B

USD-III THROUGH PARALLEL PORT

MICROTEK INTERNATIONAL INC.

B.1 INTRODUCTION

Parallel port is available from MICE-III only. It is located at the "Channel B" of the communication interface connectors located at the back panel of MICE as shown in the figure below. These connectors are housed in a factory installed Parallel-Serial Module (PSM) board which is a standard equipment of MICE-3.

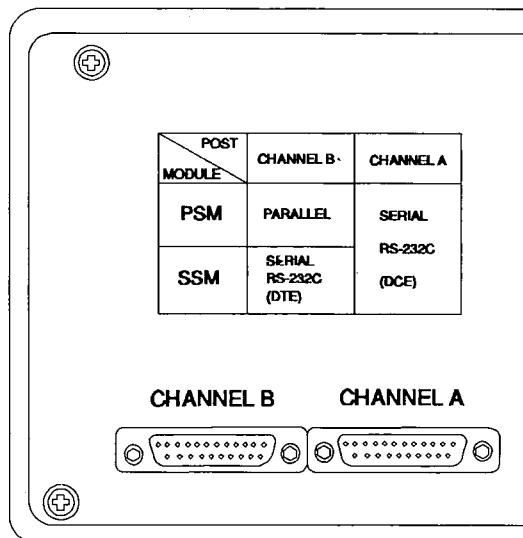


Figure B-1 MICE-III Communication Interface Connectors

B.2 INTERFACING THROUGH PARALLEL PORT

The parallel port on the PSM must connect to an appropriate interface card installed in either IBM PC/AT/XT/PS2, NEC PC-9801 or compatibles. Appropriate interface card for each model of host computer is summarized in the following table:

PARALLEL PORT

| | | | |
|----------------|-----------------------------|-----------|-------------|
| HOST COMPUTER | IBM PC/AT/XT or compatibles | IBM PS/2* | NEC PC-9801 |
| INTERFACE CARD | MS-PCE** | MS-MCE | MS-JPC |

* Applies to Models 50, 60, 70 and 80 only. For Model 30, use MS-PCE.

** Also applicable to IBM PS/2 Model 30. Unless otherwise specified, this board (MS-PCE) is factory installed with MICE.

Table B-2 Required Interface Card for Parallel Port Communication

B.2.1 IBM-PC/XT/AT MS-PCE Parallel Interface Card Installation and Setup

The MS-PCE Interface card is required for communications between IBM PS/2 Model 30, IBM PC/XT/AT (or compatible) and parallel port (PSM Channel B) of MICE-III. It is a half-card that plugs into any expansion slot except the eighth in the PC or PC/XT, or in the 8-bit bus slot for the PC/AT. The 8-bit parallel port on the PSM terminates in a DB-25P (male) D-shell connector at the back of the MICE-III. Its pin assignment and signal timing characteristics are shown in Table B-3 below.

| PIN | SIGNAL | DIRECTION | DESCRIPTION |
|-----|-------------------|---------------|--|
| 1 | GND | ----- | Signal ground |
| 2 | SEND BUFFER FULL | from MICE-III | A "LOW" signal indicates that the MICE-III has already sent data to the outgoing mailbox. (The host computer then uses the READ DATA signal to fetch it). |
| 3 | RECV BUFFER EMPTY | from MICE-III | A "LOW" signal indicates that the incoming mailbox is empty and the host computer can send the next data transmission to MICE-III. |
| 4 | READ DATA | to MICE-III | A "LOW" signal enables data in the outgoing mailbox to be available on the data bus. |

| PARALLEL PORT | | | |
|----------------------|------------|----------------|--|
| 5 | WRITE DATA | to MICE-III | A "LOW" signal enables data on the data bus to be latched into the incoming mailbox. |
| 6 | DATA 8 | Bi-directional | These signals interface the data bus. Data 8 is the most significant bit (MSB). |
| 7 | DATA 7 | Bi-directional | |
| 8 | DATA 6 | Bi-directional | |
| 9 | DATA 5 | Bi-directional | |
| 10 | DATA 4 | Bi-directional | |
| 11 | DATA 3 | Bi-directional | |
| 12 | DATA 2 | Bi-directional | |
| 13 | DATA 1 | Bi-directional | |
| 14-15 | GND | ----- | |
| 16-25 | not used | | |

Table B-3 Pin Assignment and Signals for Parallel Interface

Refer to the host computer's manual for instructions on installing auxiliary cards. Port selection and definition are discussed below:

1) Port Selection

Two adjacent I/O ports (data and status) are used as mandatory defaults on the interface card to facilitate parallel communications. Switches 3 to 8 on the DIP switch are used to specify the six Most Significant Bits (A9-A4) of the first I/O port address. The card is shipped from the factory with switches 3 to 7 on and switch 8 off, which sets the port address to be used for parallel data communications at 200H and 201H.

To set the software tool to use I/O port addresses which correspond with the switch setting, the user should specify the parallel port option value in the installation file of the tool (e.g., "P:" option in the USD-III installation file, "USD3.INS").

Address bits A3-A1 are not decoded by the interface card so the actual address space is 16 I/O ports. Data port or status port selection is indicated by bit A0. The data port is activated if A0 is low (logic 0); and the status port is activated if A0 is high (logic 1).

PARALLEL PORT

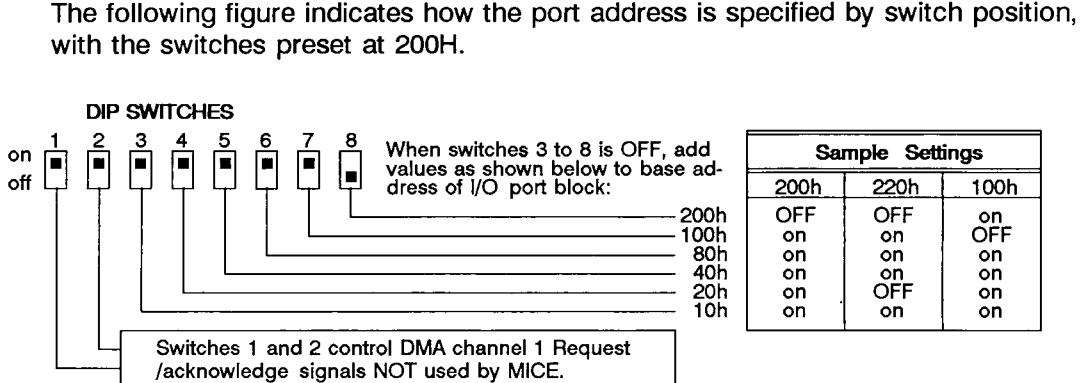


Figure B-4 MS-PCE Port Selection Example

NOTE

- Make sure the port address does not conflict with any other expansion card residing in the system. For example, the game control adapter on the AST multifunction card must be disabled since it also uses port address 200H.**
- Certain port addresses are reserved for special expansion cards or host-system functions and cannot be used for MICE I/O; check the I/O address map in the manual for the host computer before selecting an address.**
- The address selected on the card must be followed by 15 other currently unused addresses.**
- The MICE does not make use of the DMA request and acknowledge lines controlled by switches 2 and 1; the presence of the MS-PCE will not affect the functioning of any other expansion card which uses DMA channel 1.**

2) Port Definition

One Data Port and one Status Port can be defined on the MS-PCE interface card:

- The first even address port specified (e.g. 200H) is used as a bi-directional read/write Data Port.**

- b) The first odd address port specified (e.g. 201H) is used as a read-only Status Port. The first status bit lets the IBM-PC know when data in the Data Port can be read; and the second status bit lets it know when the Data Port is ready to accept data from the IBM-PC.

Status Port bit assignment is as follows:

Status Port bit assignment:

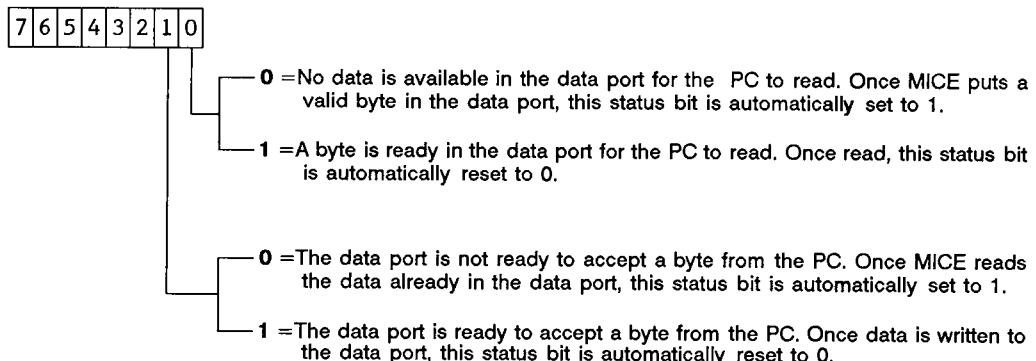


Figure B-5 MS-PCE Status Port Bit Assignment

B.2.2 IBM PS/2 MS-MCE Parallel Interface Card Installation and Setup

The MS-MCE Interface card is required for high-speed bi-directional "MAILBOX" communication between IBM PS/2 micro-channel and parallel port (PSM Channel B) of MICE-III. The card can be installed into any expansion slot of the PS/2 (Model 50/60/80). Refer to "MS-MCE Interface Card Operation Manual" and the host computer's manual for instruction on auxiliary card installation.

After performing the MS-MCE setup diagnostic with file "@60AC.ADF", initiate PS/2 parallel communication port address setting with file "MCE.EXE" by executing the following command. Note that the above mentioned files are provided with the USD-III package:

>MCE [port-address]

Where [port-address] is the I/O port address setting to be used for parallel data communication. If no port address is specified, setting will default at 200H.

PARALLEL PORT

B.2.3 NEC PC-9801 MS-JPC Parallel Interface Card Installation and Setup

The MS-JPC interface card provides bi-directional "MAILBOX" communication between MICE-III and NEC PC-9801 series computer. It can be plugged into any expansion slot of the PC-9801. Refer to the computer manual for expansion card installation.

1) Port Selection

MS-JPC interface card provides two DIP switches from which I/O port (data and status) address is selected. The I/O port address range may be specified from C0 to FF (hexadecimal).

2) Port Definition

The DIP switch U4 is the data port and the DIP switch U5 is the status port. These DIP switches are defined as follows:

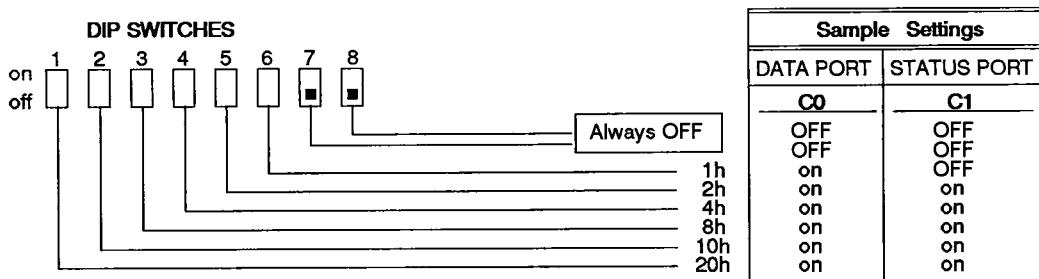


Figure B-6 MS-JPC DIP Switches U4 and U5 Port Selection Example

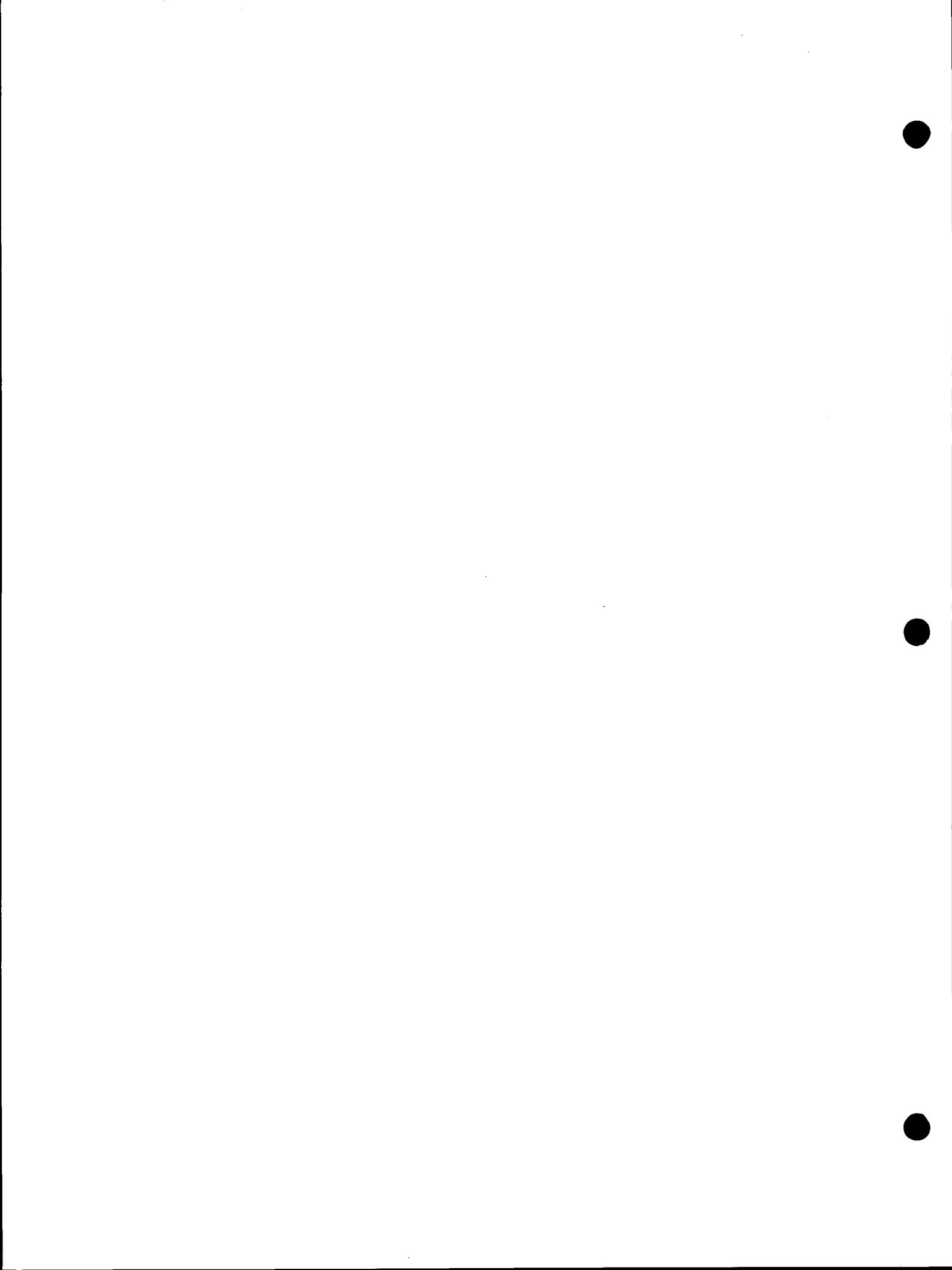
3) Port Bit Assignment

Status Port bit assignment is the same with MS-PCE (Figure B-5).

NOTE

The Data Port address and Status Port address can be set at any address between C0 and FF, but both ports should not have the same address value at the same time.

When using USD-III with NEC PC-9801 through MS-JPC card, "P: port-address" option should be specified in the USD-III's "USD3.INS" file, e.g., the port-address as data port, and the adjacent port address as status port. For example; setting "P: C0" means the data port is C0, and the status port is C1. The switches on MS-JPC should be set to correspond with the value specified in "P:" option.



Appendix C

USD-III LIMITED WARRANTY: Service

MICROTEK INTERNATIONAL INC.

Microtek International Inc. (Microtek) warrants-

Universal Symbolic Debugger - Three or USD-III

(the Product) and the user's manual for the Product to be free from physical defects for a period of twelve (12) months from the date of the original retail purchase. This warranty applies only to the original retail purchaser who bought these Products from an authorized **Microtek** representative. This warranty is void if the Product is damaged by improper or abnormal use or by accident, if the Product is altered or modified in any way, or if any attempt is made to repair the Product without authorization from **Microtek**.

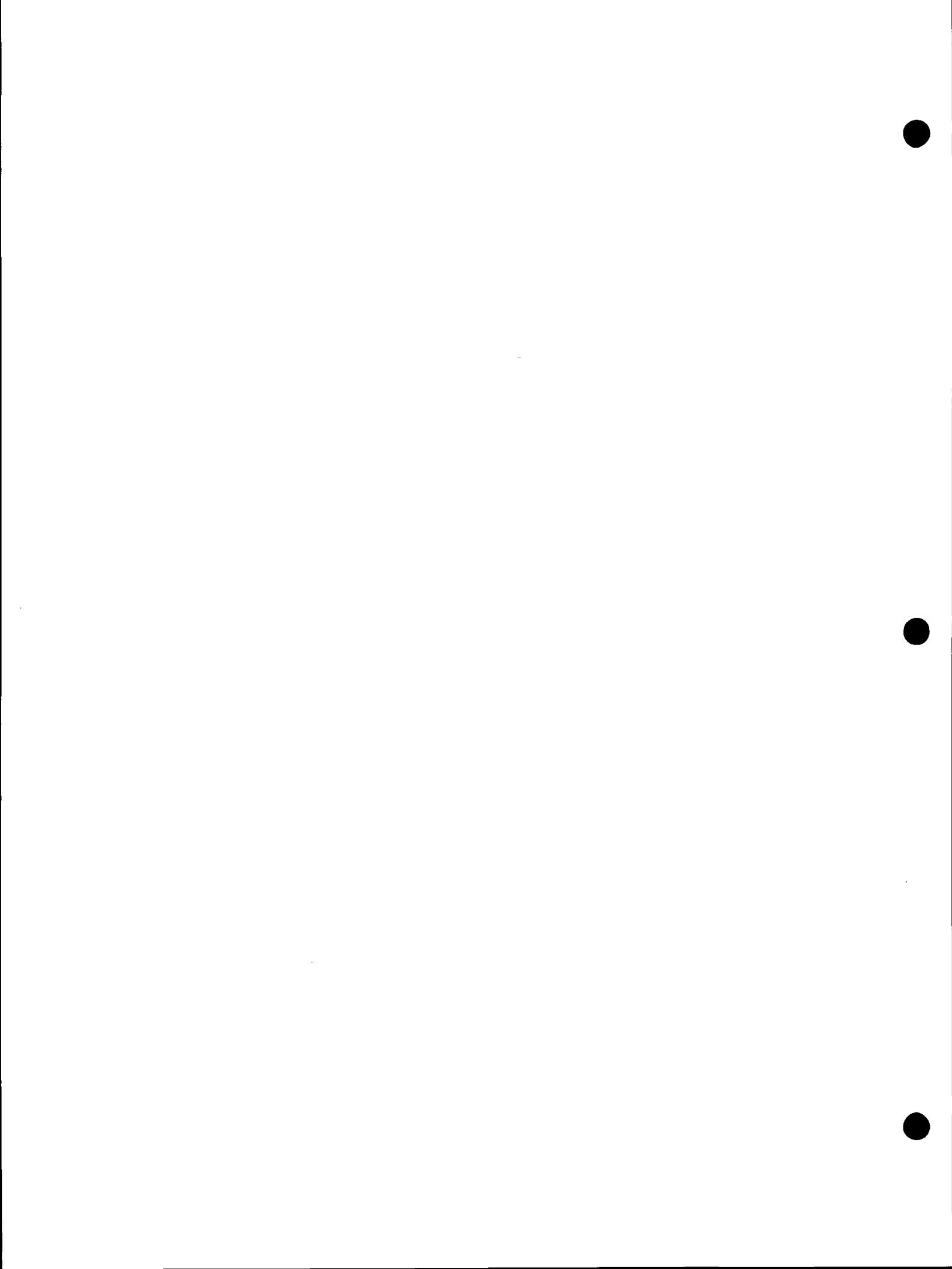
If you find a Product to be defective, contact your authorized **Microtek** representative or **Microtek**. When you receive authorization to do so, return the Product as directed; include proof of purchase and purchase date. **Microtek** will correct physical defects in Products under warranty at no charge to you by repairing or, at its option, replacing such Products. THIS IS THE SOLE AND EXCLUSIVE REMEDY AVAILABLE FOR BREACH OF WARRANTY OR UNDER ANY OTHER LEGAL THEORY WITH RESPECT TO **Microtek** PRODUCTS. Product repairs not covered by warranty, and Product updates, are provided at a set rate.

ALL OTHER WARRANTIES AND REPRESENTATIONS, ORAL OR WRITTEN, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, ARE EXCLUDED AND DO NOT APPLY. Except as set forth in this limited warranty, the purchaser assumes the risk as to these Products' quality, accuracy, design, and performance.

It is solely the purchaser's responsibility to determine the suitability of these Products for each particular application. **Microtek** Products are in all events not suitable, and are not authorized, for use in connection with life support devices or systems, or in other devices or systems potentially injurious to life or health.

IN NO EVENT WILL **MICROTEK** BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY BREACH OF WARRANTY OR UNDER ANY OTHER LEGAL THEORY, even if advised of the possibility of such damages. **MICROTEK** is thus not liable for lost profits or goodwill; downtime; damage or destruction of any program, data, equipment, or other property; costs of recovering, reprogramming, or reproducing any program, data, or equipment; personal injury or loss; or any other damages.

Except as required by law, no representative, agent, or employee of **Microtek** is authorized to commit **Microtek** to warranties, representations, or obligations inconsistent with or in addition to those set forth in this limited warranty.



INDEX

MICROTEK INTERNATIONAL INC.

:label; 4-26
 "HELP" SUMMARY; 1-17
 <\>; 4-17
 %0 - %9; 4-18
 2500AD Microtek Symbol Format; A-2

A

Address Query command; 5-67

B

Backward Trace command; 5-55
 Baud-rate; 2-3
 BELL; 4-19
 Break On Read command (*for LAM-II only*); 5-39
 Break on Read Before Write; 5-39
 BREakpoint; 4-2

C

Change Dialog Boxes Colors; 5-76
 Checksum command; 5-22
 Choice List; 1-9
 CLEar; 4-3
 Clear Monitor; 4-37
 Clear Software Breakpoint; 4-3
 Close Log/Jou command; 5-16
 Close Log/Journal File; 5-16
 Code-Coverage Test; 5-89
 COM1 or COM2; 2-3
 COMMAND DESCRIPTION; 4-1
 Command Display Control; 4-20
 COMMAND EXAMPLES:

BREakpoint; 4-2
 CLEar; 4-3
 CReate; 4-5
 DElete; 4-6
 DOS; 4-7
 DOUble; 4-8
 DOWload; 4-10
 FLoat; 4-12
 HElp; 4-13
 HIStory; 4-14
 INCLude; 4-15
 IStep; 4-31
 JOurnal; 4-32

LOAD; 4-33
 LOG; 4-34
 MODULE; 4-35
 MONitor; 4-36
 NOMon; 4-37
 OPTIONS; 4-38
 QUERY; 4-39
 QUIT; 4-40
 SOFTkey; 4-41
 SRecall; 4-43
 SSave; 4-44
 SYMbol; 4-45
 UPDate; 4-46
 UPload; 4-47
 URECall; 4-49
 USAVe; 4-50

COMMAND GROUPS:

Command Processing Utility; 3-6
 Data Monitor Commands; 3-2
 File Management Utility Commands; 3-2
 Floating Point Commands; 3-3
 Software Breakpoint Commands; 3-4
 Symbol Processing Commands; 3-4
 Viewport Processing Command; 3-5
 Command History; 4-14
 Command Processing Utility; 3-6
 COMMANDS SUMMARY; 3-1
 Comment String; 4-29
 Communication Interface; 1-2
 Compare command; 5-23
 Conditional Execution; 4-24
 CONVENTIONS OF NOTATION; 3-1
 CONVERTER PROGRAM; A-1
 Copy command; 5-24
 Coverage command (*for LAM-II only*); 5-89
 Create command; 5-61
 Create Symbol; 4-5, 5-61
 CReate; 4-5
 Cycle; 5-35

D

Data-bit; 2-3
 Data Monitor Commands; 3-2, 5-30
 Decision in Case of Error; 4-21
 Define SoftKey; 4-41
 Define Softkey command; 5-73

INDEX

Define SoftKey Macro; 5-73
DELete; 4-6
Delete command; 5-62
Delete Symbol; 4-6, 5-62
Dialog Color command; 5-76
Directory Module; 4-35
Disassembly command; 5-25
Display/Set/Clear Breakpoints; 5-43
DOS; 4-7, 5-3
DOuble; 4-8
Double Floating Point; 4-8
DOWnload; 4-10
Download command; 5-4
Download File; 4-10, 5-4
Download Format; 1-2
Dump command; 5-26
Duration command; 5-85

E

ECHO; 4-20
"Emulation" Pull-Down Menu Commands; 5-34
Enable/Disable Symbol/Log/Journal/S/W Breakpoint; 5-69
Enable/Disable Viewport Update; 4-46, 5-71
Entry command; 5-83
ERROR; 4-21
Event command; 5-43
Execute Command File; 4-15, 5-10
EXECUTING COMMANDS FROM MENUS; 5-1
Execution Time Analysis; 5-87

F

File Management Utility Commands; 3-2
"File" Pull Down Menu Commands; 5-3
Fill command; 5-27
FLoat; 4-12
Floating Point Commands; 3-3
Forward Trace command; 5-56

G

GET; 4-22
GETTING STARTED; 2-1
Go; 5-36
GOTO; 4-23

H

Halt; 5-36
Handshake code; 2-3
HElp; 4-13
HIStory; 4-14
Host Computer; 1-2

I

IAR System Format; A-3
IBM PS/2 MS-MCE Parallel Interface Card; B-5
IBM-PC/XT AT MS-PCE Parallel Interface Card; B-2
IEEE Format for Binary Representation; 3-3
IF / ELSE; 4-24
INClude; 4-15, 25
Include command; 5-10
Include User Symbols from File; 4-43
INSTALLATION; 2-1
Instruction Step; 4-31
Intel OMF Format; A-4
Interface Cards%; 2-4
 MS-MCE; 2-4
 MS-PCE; 2-4
 MS-JPC; 2-4
 PSM; 2-4
 SSM; 2-4
INTERFACING THROUGH PARALLEL PORT; B-1
Intermetrics Object Language Format; A-4
INTRODUCTION; 1-1
INVOKING USD-III; 2-5
IStep; 4-31, 5-34
Istep Forever; 5-35
Istep Over; 5-35
Istep Over Forever; 5-35

J

JOurnal; 4-32
Journal File command; 5-14

L

Label to be Referred to by "GOTO" Sub-Command; 4-26
List command; 5-57, 5-59
List Symbol; 4-45
List Trace Buffer; 5-57
LOAD; 4-33
Load command; 5-60

Load Symbol Table; 4-33, 5-60
LOG; 4-34
Log Commands; 5-12
Log Command File; 4-34, 5-12
Log Output File; 4-32
Log Output and Create Output File; 5-14
Loop Block; 4-27
LOOP/LOOPEND; 4-27
"LSA" Menu Bar Command; 5-94

M

Map command; 5-19
Memory Checksum; 5-22
Memory Compare; 5-23
Memory Copy; 5-24
Memory Disassembly; 5-25
Memory Dump; 5-26
Memory Fill; 5-27
Memory Map; 5-19
"Memory" Pull-Down Menu Commands; 5-21
Memory Search; 5-28
Memory Test; 5-29
Menu Bar; 1-5
MENU COMMANDS; 5-1
Address Query command; 5-67
Backward Trace command; 5-55
Break on Read command; 5-39
Checksum command; 5-22
Close Log/Jou command; 5-16
Compare command; 5-23
Copy command; 5-24
Coverage command; 5-89
Create command; 5-61
Data Monitor command; 5-30
Define Softkey command; 5-73
Delete command; 5-62
Dialog Color command; 5-76
Disassembly command; 5-25
Dos command; 5-3
Download command; 5-4
Dump command; 5-26
Duration command; 5-85
Entry command; 5-83
Event command; 5-43
Fill command; 5-27
Forward Trace command; 5-56
Include command; 5-10
Journal File command; 5-14
List command; 5-57

Load command; 5-60
Log File command; 5-12
Quit command; 5-17
SSave command; 5-63
Search command; 5-28
SRecall command; 5-64
Symbol Query command; 5-66
Test command; 5-29
Time command; 5-87
Toggle Status command; 5-69
Trace Control command; 5-51
Trace Control command (*for LAM-II only*); 5-48
Trigger Level command (*for LAM-II only*); 5-44
Update Viewport command; 5-71
Upload command; 5-7
Urecall command; 5-75
Usav command; 5-74
Version command; 5-3

MENU COMMAND GROUPS:

"Emulation" Pull-Down Menu Commands; 5-34
"File" Pull Down Menu Commands; 5-3
"LSA" Menu Bar Command; 5-94
"Memory" Pull-Down Menu Commands; 5-21
"Options" Pull-Down Menu Commands; 5-68
"Setup" Pull Down Menu Commands; 5-18
"Spa" Pull-Down Menu Command; 5-79
"Symbol" Pull-Down Menu Commands; 5-59
"Trace" Pull-Down Menu Commands; 5-36
Message Line-; 1-11
- default; 1-11
- for current command; 1-13
- with Dialog Box; 1-15
- with Pull-Down Menu; 1-14
Module; 5-59
MODULE; 4-35
Module Duration Analysis; 5-85
Module Entry Analysis; 5-83
MONitor; 4-36
MOUSE OPERATION; 1-4
Mouse Support; 1-2
MRI Symbol Format; A-5
MS-MCE Interface card; 2-4, B-5
MS-JPC Interface card; 2-4, B-6
MS-PCE Interface card; 2-4, B-2

N

NEC PC-9801 MS-JPC Parallel Interface Card; B-6
Nest Command File; 4-25
NOMon; 4-37

INDEX

Non-volatile RAM (NOVRAM); 5-19

O

OM/LM Format (SYSROF); A-6
ON-LINE <F1> "HELP" SUMMARY; 1-17
Operating System; 1-2
Optional Features; 4-38
OPTIONS; 4-38
"Options" Pull-Down Menu Commands; 5-68

P

PARALLEL PORT; B-1
Parallel Port Communication; B-2
Parallel Port Setup (*MICE-III* only); 2-4
Parity; 2-3
Pin Assignment and Signals for Parallel Interface; B-3
Pop-Up Dialog Box; 1-9
Port status setting; 2-3
Pre-defined Sub-command:
 \; 4-17
 %0 - %9; 4-18
 BELL [count]; 4-19
 ECHO [ON|OFF]; 4-20
 ELSE condition command; 4-24
 ERROR {ABORT|CONTINUE}; 4-21
 GET str; 4-22
 GOTO label; 4-23
 IFI[NOT] condition command; 4-24
 INCLUDE filename [(arg0,...,arg9)]; 4-25
 :label; 4-26
 LOOP; 4-27
 LOOPEND [count]; 4-27
 PAUSE [str|count]; 4-28
 REMARK str; 4-29
 TICK [count]; 4-30
Prompt for Keyboard Input; 4-22
PSM; 2-4, B-1
Pull-Down Menus; 1-7, 1-8

Q

QUEry; 4-39
Query for Symbol Name; 5-66
Query for Address; 5-67
QUIT; 4-40
Quit USD-III; 2-6, 4-40, 5-3, 5-17
Quit command; 5-17

R

Recall Symbols from File; 5-64
Recall Window Status and Softkey Definitions; 4-49,
 5-75
Redirect Flow of Command Execution; 4-23
REMark; 4-29
Ring the Bell; 4-19
RS-232C Serial Port Setup; 2-3

S

S/W Breakpoint command; 5-37
Save Current Window Status and Softkey Definitions;
 5-74
Save Symbols to File; 4-44
Save Window Status and Softkey Definitions; 4-50
SCREEN LAYOUT; 1-5
 Message Line; 1-11
 Choice List; 1-9
 Dialog Box; 1-9
 Pull-Down Menu; 1-7
 Menu Bar; 1-5
 Viewports; 1-6
Search command; 5-28
Set/Clear Monitor Variable; 5-30
Set/Clear Software Breakpoint; 5-37
Set Monitor Variable; 4-36
Set Software Breakpoint; 4-2
Set Time Delay; 4-30
Set Trigger Level; 5-44
SETUP ON MS-DOS/PC-DOS HOSTS; 2-2
"Setup" Pull Down Menu Commands; 5-18
Single Floating Point; 4-12
SOFTkey; 4-41
Software Breakpoint Commands; 3-4
SOFTWARE BREAKPOINT IMPLEMENTATION; 2-8
SPA Applications; 5-81
"Spa" Pull-Down Menu Command; 5-79
SRecall; 4-43
SRecall command; 5-64
SSave; 4-44
SSave command; 5-63
SSM; 2-4
Substitute for <Esc> in Command File; 4-17
SYMBOL; 4-45
Symbol/Address Query; 4-39
Symbol Processing Commands; 3-4
"Symbol" Pull-Down Menu Commands; 5-59
Symbol Query command; 5-66

Symbol Table Size; 1-2
System Overview; 1-1

T

Temporary Exit to DOS; 4-7
Test command; 5-29
TiCK; 4-30
Time command; 5-87
Toggle Status command; 5-69
Tool Chain; 1-3
Trace Control command (*for LAM-II only*); 5-48
Trace Control Settings; 5-48, 5-51
Trace with Forward Count; 5-56
Trace with Backward Count; 5-55
Trace Control command; 5-51
Trigger Level command (*for LAM-II only*); 5-44
"Trace" Pull-Down Menu Commands; 5-36

U

UPDate; 4-46
Update Viewport command; 5-71
UPload; 4-47
Upload File; 4-47, 5-7
Upload command; 5-7
URECall; 4-49
Urecall command; 5-75
USAve; 4-50
Usav command; 5-74
USD-III Command Help; 4-13
USD-III Files; 1-3
USD-III menu commands; 5-1

V

Variables to be Replaced; 4-18
Version; 5-3
Viewport Processing Commands; 3-5
Viewports; 1-6
CODE; 1-6
COMMAND; 1-6
DATA; 1-6
REAKPOINT; 1-6
REGISTER; 1-6
STACK; 1-6
TRACE; 1-6

W

WARRANTY; C-1

