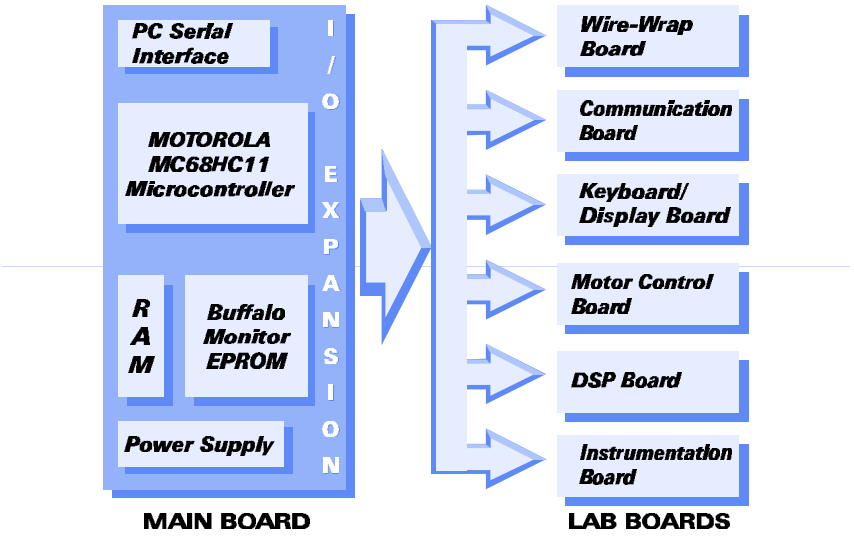
EZ-MICRO Tutor™

"State-of-the-Art Teaching Tools"



Advanced Microcomputer Systems

The EZ-MICRO product is published and marketed by
Advanced microcomputer Systems, Inc.

Specifications are subject to change without notice.

July 1996, 1st Printing

Advanced Microcomputer Systems, Inc.
1460 SW 3rd Street
Pompano Beach, FL 33069
Phone (954) 784-0900
Fax (954) 784-0904

E-Mail: ams@gate.net

Worldwide Web: http://www.gate.net/~ams/

Features of the EZ-Micro Manager Software

The EZ-MICRO Manager software is used to communicate (through a host computer or terminal's serial port) with the EZ-MICRO CPU-11 microprocessor board (which utilizes a Motorola MC68HC11D0 microprocessor).

The EZ-MICRO Manager software allows you to modify and display on-board user data/program memory, disassemble program memory (to the screen or dump to a file), assemble source code and download corresponding hex files, execute user programs, set breakpoints and display "watch" variables, modify and display CPU register contents and display the contents of user memory "graphically" on the screen.

Running the EZ-MICRO Manager in the demo mode will allow you to evaluate the features and the ease of operation of using this "menu driven" monitor program. Unlike other "monitor" programs that require you to memorize and type numerous commands, the EZ-MICRO Manager software has all of the commands listed for you in categorized pull down menus.

Note: Certain options in the pull down menus may be "*grayed out*" and can not be selected when running in the demo mode.

System Requirements

In order to run the EZ-MICRO Manager software, the following equipment is required:

- **IBM personal computer (or compatible)**
- **MS-DOS (version 3.0 or later)**
- **Minimum 640K RAM memory**
- **EGA or VGA graphics adapter card**
- **RS-232 serial interface port**
- **A mouse is suggested, but is not required**
- **A hard disk is suggested, but is not required**

Introduction to the EZ-Micro Manager & EZ-Micro CPU-11

The EZ-MICRO Manager demo, will allow you to evaluate most of the features of the EZ-MICRO Manager software. Some of the options in the pull down menus will be "grayed out" and will only function when the EZ-MICRO CPU-11 microprocessor board is connected to the serial port of the host computer.

The EZ-MICRO CPU-11 utilizes a Motorola MC68HC11D0 device which is an advanced single-chip CPU with 192 bytes of on-chip memory and peripheral functions. To demonstrate the capabilities of this CPU, the EZ-MICRO CPU-11 microprocessor board was designed along with a monitor/debugging program called BUFFALO (Bit User Fast Friendly Aid to Logical Operations). This monitor program is contained in the EPROM that is external to the CPU. Note: The complete source code (BUFFALO.ASM) and list file (BUFFALO.LST) of the BUFFALO monitor is provided with the EZ-MICRO CPU-11 microprocessor board.

The EZ-MICRO CPU-11 allows the user to debug user code under control of the EZ-MICRO Manager software. User code is assembled (into a Motorola S-Record format) using the EZ-MICRO Manager and is then downloaded to the EZ-MICRO CPU-11's RAM via the host computer's RS-232 serial port. Also, a single line assembler/disassembler can be used to input code directly into user RAM. The EZ-MICRO Manager is then used to execute/debug the assembled user code.

Features of the EZ-Micro CPU-11 Microprocessor Board

The EZ-MICRO CPU-11 provides a low cost method for debugging and evaluating Motorola's MC68HC11D0 microprocessor. The MC68HC11D0 is a high-performance micro controller unit that is based on the MC68HC11E9 design. The MC68HC11D0 chip offers high speed, low power consumption and multiplexed buses capable of running up to 3 Mhz. The EZ-MICRO CPU-11 microprocessor board is shown below:

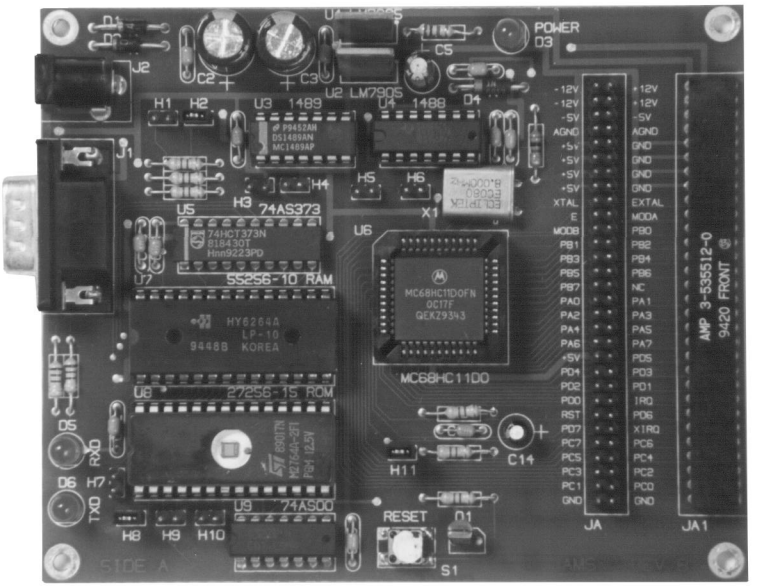


Figure 1: EZ-MICRO CPU-11 Microprocessor Board.

The MC68HC11D0 is also an excellent economical alternative device for applications that require the HC11 CPU but where fewer peripheral functions and less memory are required. Refer to the Motorola MC68HC11D0 CPU data sheet (document #BR777/D) for additional device information.

An optional wire-wrap section can be provided with the EZ-MICRO CPU-11 to provide an area for custom interfacing to the CPU. The wire-wrap section can be detached from the EZ-MICRO CPU-11 so that a new wire-wrap section or one of the other EZ-MICRO CPU-11 interfacing projects can be connected. This allows for the same EZ-MICRO CPU-11 microprocessor board to be used over and over again on many different projects.

The EZ-MICRO CPU-11 also includes the following features:

- MC68HC11D0 CPU in a 44 pin PLCC package.
- Detachable wire-wrap area for custom interfacing (optional).
- Detachable keypad/seven segment display lab board (optional).
- Detachable ADC/DAC lab board (optional).
- 192 Bytes On-Chip RAM.
- RS-232C compatible I/O interface port.
- Single input power supply to provide all voltage requirements.

- Real-Time interrupt circuit.
- 16-Bit Timer System - 3 Input Capture (IC) Channels/4 Output Compare (OC) Channels. Additional Channel is software selectable as either Fourth IC or Fifth OC.
- Computer Operating Properly (COP) Watchdog System.
- Synchronous Serial Peripheral Interface (SPI).
- Asynchronous Nonreturn to Zero (NRZ) Serial Communications Interface (SCI).
- 8-Bit Pulse Accumulator.
- 26 Input/Output (I/O) Pins.

EZ-Micro CPU-11 Specifications

The following table lists the specifications of the EZ-MICRO CPU-11 microprocessor board:

Characteristic	Specification
CPU	MC68HC11D0
Terminal I/O Port	RS-232C compatible
Temperature:	
Operating	+25 Degrees C
Storage	-40 to +85 Degrees C
Relative Humidity	0 to 90% (non-condensing)
Dimensions:	
Width	4.050 in. (10.287 cm)
Length	4.925 in. (12.510 cm)
Wire-wrap Area (optional):	
Area	Approximately 11.1 sq. in
Holes	31 x 38 (0.100 in. centers)

TABLE 1: EZ-MICRO CPU-11 Specifications.

Demo Software Installation

To install the EZ-MICRO Manager software DEMO onto your hard drive, insert the DEMO disk that was provided into your floppy drive. Change the current directory to your floppy drive by typing the following at the prompt:

A: (and press the <ENTER> key)

Note: If your floppy drive is Drive B, then substitute "B:" in the above command. Next, type "INSTALL" and press the <ENTER> key to begin the installation. You will be further instructed on what to do.

To run the demo software, make sure that the current directory is the directory where the demo manager software is located. Refer to the following sections for instructions on starting the demo.

Starting the EZ-Micro Manager Demo

Several files (including several sample assembly language routines) will be included on the EZ-MICRO Manager demo disk. A description of each file is shown below for your convenience:

File Name	Description of file.
EZMICRO.EXE	EZ-MICRO Manager software.
AS11.EXE	Optional 68HC11 assembly language command line assembler.
PLOT1.ASM - PLOT4.ASM	Sample "waveforms" that can be assembled, downloaded and plotted to the screen or to a file.
LAB1.ASM	Sample assembly language program that initializes an array in memory.
MEM_TEST.ASM	A non-destructive memory test routine.
NAME.ASM	Sample program that prompts the user to enter a name and then "flashes" the name on the screen.
LEDDEMO.ASM	Test program for the Keypad/Display Lab board to count from 000 to 999 and displays on LEDs.

KEYDEMO.ASM	Test program for the Keypad/Display Lab board to display the number of the key pressed on the LEDs and to the screen.
LEDS.ASM	Test program for one of the interfacing LED display boards.
CHAL_LED.ASM	Test program for one of the interfacing LED display boards.
A2D_D2A.ASM	Can be used to digitally convert and store (into user RAM) an analog input signal. This digitally stored signal can be "played-back" and outputted through the DAC logic.
SQUARE.ASM	This program is used to generate a "square" wave signal on the ADC/DAC Lab board.
SAWTOOTH.ASM	This program is used to generate a "sawtooth" wave signal on the ADC/DAC Lab board.
TRIANGLE.ASM	This program is used to generate a "triangular" wave signal on the ADC/DAC Lab board.

If you wish to run the EZ-MICRO Manager demo, you must first change the current directory to the directory where the demo has been installed. This can be done by typing the following at the prompt:

C: (and press the <ENTER> key)
CD\EZMICRO (and press the <ENTER> key)

Note: "D:" may be substituted in the above command if you have installed the EZ-MICRO Manager demo software on Drive D. After changing to the correct directory, type the following at the prompt:

EZMICRO DEMO (and press the <ENTER> key)

and the software will be started in the demo mode. Note: If you are evaluating the EZ-MICRO CPU-11 microprocessor board, then before starting the EZ-MICRO Manager software, press the master reset button (S1) on the EZ-MICRO CPU-11. This will ensure that the EZ-MICRO Manager software will communicate properly with the EZ-MICRO CPU-11 hardware. After pressing the master reset button, type "EZMICRO" instead of "EZMICRO DEMO" to start-up the software in the "normal" mode.

How Does the EZ-Micro Manager Software Work ?

The EZ-MICRO Manager software makes it possible to communicate directly to the EZ-MICRO CPU-11 (Motorola's MC68HC11D0 microprocessor) using any IBM compatible "host" computer or terminal that has at least one asynchronous serial port.

All of the commands that are issued to the EZ-MICRO CPU-11 microprocessor board are selected from various pull down menus located across the top of the EZ-MICRO Manager presentation screen (Figure 2). Therefore, it is not necessary to memorize and type numerous commands in order to communicate to the CPU using this software.

The EZ-MICRO Manager presentation screen will be displayed when the program is initially started and is shown below.

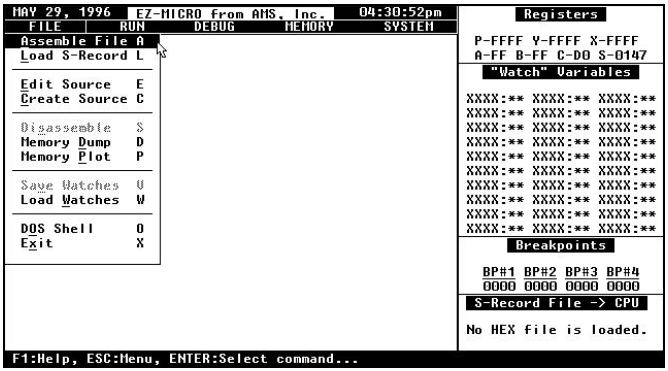


Figure 2: EZ-MICRO Manager Presentation Screen.

The EZ-MICRO Manager presentation screen is divided into eight areas.

■ The **"Title Bar"**, located at the top of the screen, displays the current date and time and also the title of this software.

■ The **"Action Bar"**, located just below the Title Bar, contains the various pull down menus where the commands that are issued to the EZ-MICRO CPU-11 microprocessor board can be selected. The pull down menus are shown in the following figure:

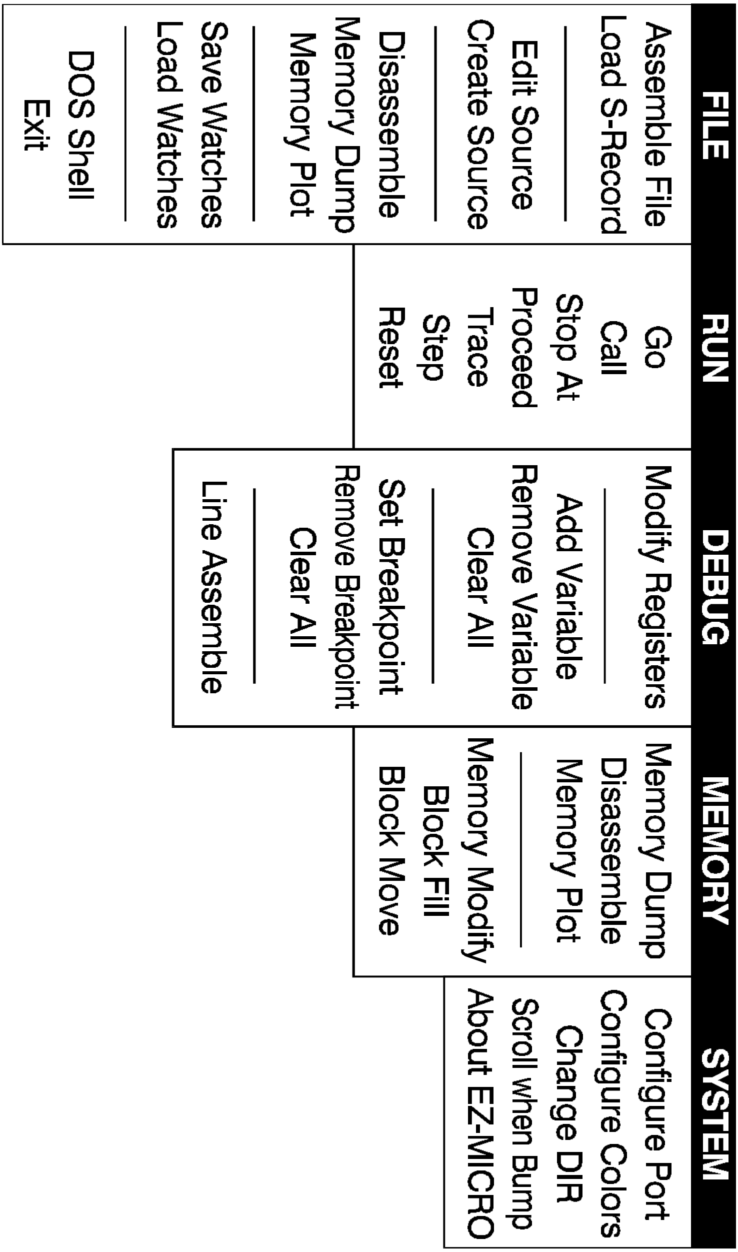


Figure 3: EZ-MICRO Pull Down Menus.

One of the pull down menus will normally be pulled down. If a menu is not currently pulled down, you can do one of the following:

* Press the ESC key.
* Point at the menu heading with the mouse.
* Press the key combination ALT plus the first letter of the menu's heading.

Notice that in all three cases a menu will be pulled down. To access an adjacent menu, simply press the right or left arrow keys on the keyboard or point at the menu heading with the mouse.

A menu bar is located within the pull down menu and is used to select one of the commands in the menu. The menu bar can be moved up or down by pressing the up or down arrow keys on the keyboard. Once the menu bar is positioned on the desired command, press the ENTER key to select that command. If you are using a mouse, point at the command and click the left mouse button to select the command. Listed to the right of each command in the pull down menus are "*hot keys*" which allow you to select a command (in the current pulled down menu) by simply pressing its hot key on the keyboard.

■ The **"Registers"** display window is located in the upper right corner of the EZ-MICRO Manager presentation screen. This window is used to display the current EZ-MICRO CPU-11 register contents. The EZ-MICRO CPU-11 registers are: the Program Counter (P), Y-index (Y), X-index (X), A-accumulator (A), B-accumulator (B), Condition Code (C) and Stack Pointer (S).

■ The **"Watch Variables"** display window is located below the Registers display window. This window is used to monitor the contents of several memory locations at once while running/debugging the assembled code.

■ The **"Breakpoints"** display window is located below the Watch Variables display window. This window is used to display the current user breakpoints that have been entered into the breakpoint address table.

■ The **"S-Record File -> CPU"** display window is located below the Breakpoints display window. This window is used to display the name of the last assembled 68HC11 hex file (also known as a S-Record file) that has been downloaded into user memory on the EZ-MICRO CPU-11 microprocessor board.

■ The **"Main Display"** window is the large area located in the left half of the EZ-MICRO Manager presentation screen. This window is used to display the EZ-MICRO CPU-11 memory contents, disassembled memory listing, trace of program execution and a graphical plot of user memory.

■ The **"Status Line"** display window is located at the bottom of the EZ-MICRO Manager presentation screen. This window is used to display the current command that has been selected and any options that may be available for that command.

Getting Help

On-line help is always available by pressing the F1 key and is context sensitive. That is, the help dialog box that appears will be concerning the current highlighted command. For example, highlight a command (using the menu bar) in one of the pull down menus and press the F1 key. A "*help*" dialog box will appear explaining how the command is to be used.

EZ-Micro Manager Commands and Tutorial

The File Menu

The FILE menu contains commands that will access "file type" operations. That is, these commands will read/write to disk memory of the "host" computer or terminal.

Assemble File

The "Assemble File" command in the FILE menu is used to assemble a Motorola 68HC11 assembly language source file into a Motorola S-Record hex file. The assembled S-Record file can be later downloaded to the EZ-MICRO CPU-11 microprocessor board and executed.

For example, select the Assemble File command from the FILE menu. Since we have not yet assembled a source file, the file name "noname.asm" will be present in the dialog box. Press the ESC key (or the right mouse button) to remove the existing name in the dialog box and press the ENTER key twice (or click the left mouse button twice) to display the available assembly language source files (*.asm).

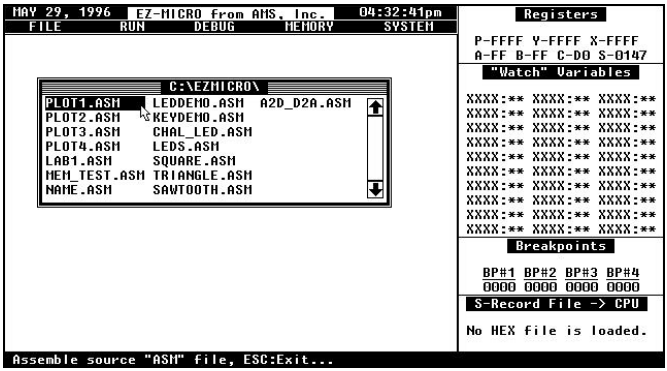


Figure 4: Assembly Language Source Files Listing.

Select the file "plot1.asm" by pointing at the file name with the mouse and clicking the left mouse button. The following dialog box will be displayed showing the options that are available for the assembler. The file, "plot1.asm", contains a sinusoidal analog signal that was sampled using ADC logic and stored digitally into memory. Press the OK button to begin assembling the file. The assembly language source file will be assembled to the file "plot1.s19". The symbol table, cross reference table and any errors that occur will be listed in the file "plot1.lst".

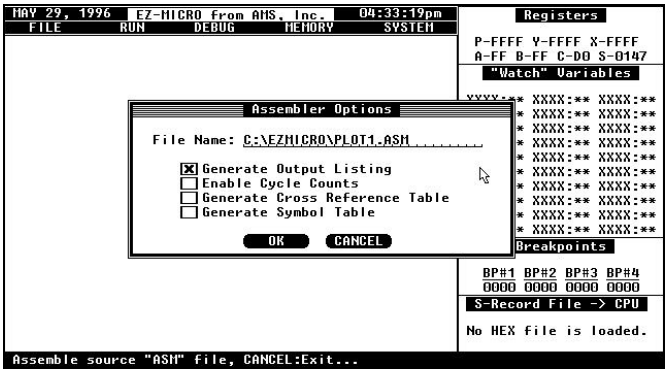


Figure 5: Assembler Options.

Load S-Record

The "Load S-Record" command in the FILE menu is used to download an assembled Motorola S-Record hex file into user memory located on the EZ-MICRO CPU-11 microprocessor board.

Note: The Load S-Record command can also be selected by pointing at the "S-Record File -> CPU" window (in the lower right corner of the screen) with the mouse and clicking the left mouse button.

For example, select the Load S-Record command from the FILE menu. Since we have not yet selected an assembled source file, the file name "noname.s19" will be present in the dialog box. Press the ESC key (or the right mouse button) to remove the existing name in the dialog box and press the ENTER key twice (or click the left mouse button twice) to display the available assembled Motorola S-Record hex files.

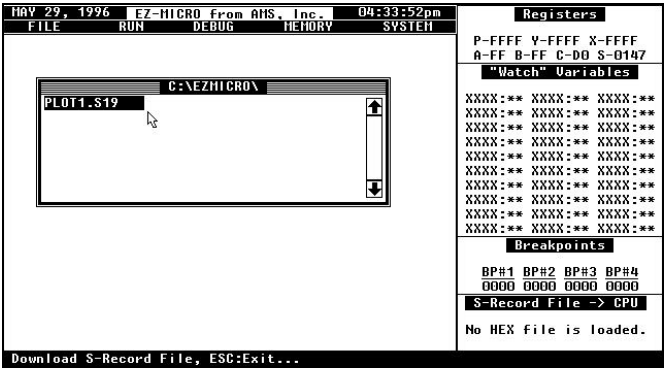


Figure 6: Assembled S-Record Hex Files Listing.

Select the S-Record file "plot1.s19" by pointing at the file name with the mouse and clicking the left mouse button. Note: The file "plot1.s19" was previously generated in the section "Assemble File". The S-Record file will be downloaded into memory on the EZ-MICRO CPU-11 and the message "S-Record successfully loaded!" will be displayed when the transfer has been completed. Also, the name of the last assembled S-Record file that has been downloaded into user memory on the EZ-MICRO CPU-11 microprocessor board will be displayed in the lower right corner of the screen (in the "S-Record File -> CPU" display window).

To view the "data" that has been downloaded into memory, select Memory Plot from the MEMORY menu. After selecting the Memory Plot command, you will be prompted to enter the starting hex address to begin the memory plot. Enter

the hex address "1000" for this example and press the ENTER key (or click the left mouse button) to accept this number. The Main Display window will expand and a plot of the contents of memory will be displayed. More information concerning the Memory Plot command will be explained later. When you have finished viewing the memory plot, press the ESC key (or the right mouse button).

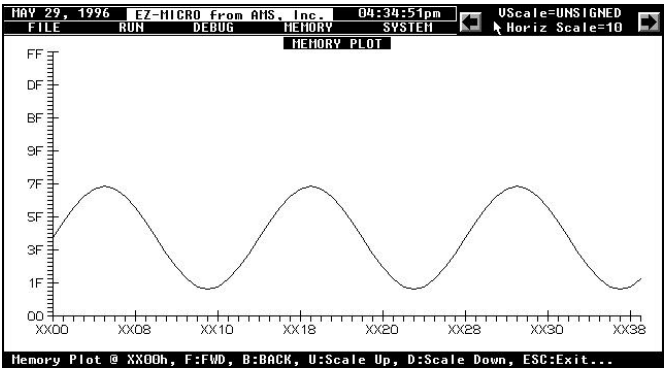


Figure 7: Memory Plot.

Edit Source

The "Edit Source" command in the FILE menu is used to edit an existing assembly language source file (or any ASCII file).

For example, select the Edit Source command from the FILE menu. Since we have not yet selected a file to edit, the file name "noname.asm" will be present in the dialog box. Press the ESC key (or the right mouse button) to remove the existing name in the dialog box and press the ENTER key twice (or click the left mouse button twice) to display the available assembly language source files.

Select the file "mem_test.asm" by pointing at the file name with the mouse and clicking the left mouse button. The Main Display window will now expand to display the assembly language source file. The editor commands can be displayed by pressing the F1 key. Press the F1 key now to display these commands. When you have finished viewing the editor commands, press the ESC key and you will return to the currently loaded assembly language source file to edit.

When you have finished viewing/editing the file, press the ESC key to exit the editor. If any changes were made to the file, you will be prompted to save these changes.

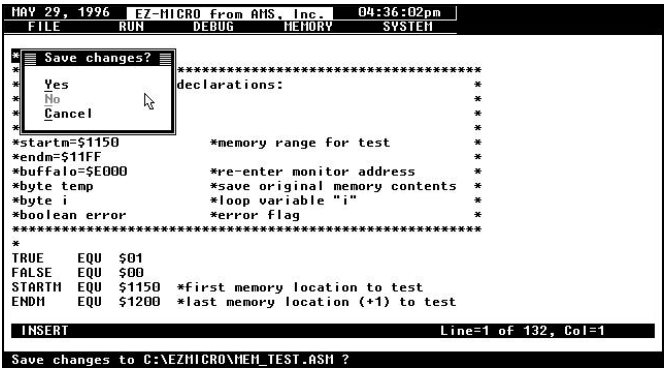


Figure 8: Edit Source File Display.

Create Source

The "Create Source" command in the FILE menu is used to create a new assembly language source file. After selecting the Create Source command from the FILE menu, you will be prompted to enter the name that will be given to this new file. When you have finished editing the new source file, press the ESC key and you will be prompted to save this newly created file.

Disassemble

The "Disassemble" command in the FILE menu is used to write the disassembled contents of memory to a DOS file. Note: This option will be "grayed out" and can not be selected while running the EZ-MICRO Manager in the demo mode.

When the Disassemble command is selected, a dialog box will appear allowing you to enter the name of the file (default file name = EZMICRO.DMP) to receive the disassembled memory listing. Also, you must enter the starting and ending address range (of memory) to be disassembled.

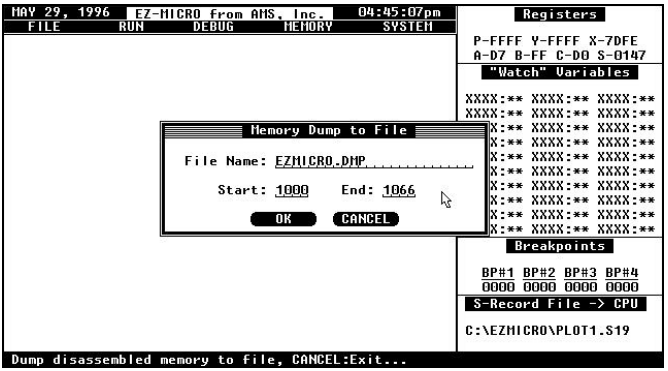


Figure 9: Disassemble Memory to a File.

The following example shows a disassembled memory listing created by the EZ-MICRO Manger software:

```
EZ-MICRO Disassembled Memory Dump
START_DISASSEMBLE_DUMP
1000 LDAA #    $00
1002 STAA      $0900
1005 LDX  #    $1150
1008 LDAA      $00,X
100A STAA      $0901
100D LDAA #    $00
100F STAA      $0902
1012 STAA      $00,X
1014 CMPA      $00,X
.
.
.
1058 PSHA
1059 LDAA      $00,X
105B CMPA #    $04
105D BEQ       $1065
105F JSR       $E3C2
1062 INX
1063 BRA       $1059
1065 PULA
1066 RTS
END_DISASSEMBLE_DUMP
```

Example 1: EZMICRO.DMP (Disassembled Memory Listing).

Memory Dump

The "Memory Dump" command in the FILE menu is used to write the hexadecimal contents of memory to a DOS file.

When the Memory Dump command is selected, a dialog box will appear allowing you to enter the name of the file (default file name = EZMICRO.HEX) to receive the hexadecimal memory listing. Also, you must enter the starting and ending address range (of memory) that will be included in this memory dump.

For example, select the Memory Dump command from the FILE menu and the following dialog box will appear.

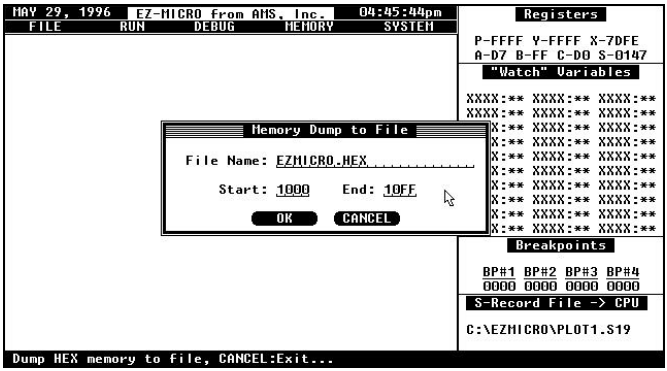


Figure 10: Hexadecimal Memory Dump to a File.

The file name "ezmicro.hex" will do for this example. Enter "1000" for the starting address and enter "10FF" for the ending address. Note: When the EZ-MICRO Manager is running in the demo mode, there are 256-bytes of memory (0000h-00FFh) available in order to make the commands in the pull down menus appear to function. When the EZ-MICRO CPU-11 microprocessor board is connected to the host computer or terminal (via the RS-232 serial port), then there will be 8k-bytes of ROM and 8k-bytes of RAM available to the user. Press the OK button to accept the parameters entered in the dialog box (Figure 11).

We can use the EZ-MICRO Manager's editor to view the results of the memory dump that was just generated. Select the Edit Source command from the FILE menu. The name of the last file that was edited, "mem_test.asm", will be displayed in the dialog box that appears. Press the ESC key (or the right mouse button) to remove this file name and enter "ezmicro.hex". Now press the ENTER key (or click the left mouse button) and the following hexadecimal listing should be displayed.

```
EZ-MICRO Memory Hex Dump
START_HEX_DUMP
XX00 4B 5A 68 73 7A 7D 7A 73 68 5A 4B 3B 2E 23 1B 19
XX10 1B 23 2E 3B 4B 5A 68 73 7A 7D 7A 73 68 5A 4B 3B
XX20 2E 23 1B 19 1B 23 2E 3B 4B 5A 68 73 7A 7D 7A 73
XX30 68 5A 4B 3B 2E 23 1B 19 1B 23 2E 3B 4B 5A 68 73
XX40 7A 7D 7A 73 68 5A 4B 3B 2E 23 1B 19 1B 23 2E 3B
XX50 4B 5A 68 73 7A 7D 7A 73 68 5A 4B 3B 2E 23 1B 19
XX60 1B 23 2E 3B 4B 5A 68 73 7A 7D 7A 73 68 5A 4B 3B
XX70 2E 23 1B 19 1B 23 2E 3B 4B 5A 68 73 7A 7D 7A 73
XX80 68 5A 4B 3B 2E 23 1B 19 1B 23 2E 3B 4B 5A 68 73
XX90 7A 7D 7A 73 68 5A 4B 3B 2E 23 1B 19 1B 23 2E 3B
XXA0 4B 5A 68 73 7A 7D 7A 73 68 5A 4B 3B 2E 23 1B 19
XXB0 1B 23 2E 3B 4B 5A 68 73 7A 7D 7A 73 68 5A 4B 3B
XXC0 2E 23 1B 19 1B 23 2E 3B 4B 5A 68 73 7A 7D 7A 73
XXD0 68 5A 4B 3B 2E 23 1B 19 1B 23 2E 3B 4B 5A 68 73
XXE0 7A 7D 7A 73 68 5A 4B 3B 2E 23 1B 19 1B 23 2E 3B
XXF0 4B 5A 68 73 7A 7D 7A 73 68 5A 4B 3B 2E 23 1B 19
END_HEX_DUMP
```

Example 2: EZMICRO.hex (Hexadecimal Memory Dump).

Note: The memory addresses will appear as "XX00" only during the demo mode. This is done since there are only 256-bytes of memory (0000h-00FFh) available when running the EZ-MICRO Manager in the demo mode. When you have finished viewing the hexadecimal memory dump, press the ESC key to exit the editor.

Memory Plot

The "Memory Plot" command in the FILE menu is used to graphically display the contents of memory in a DOS file.

When the Memory Plot command is selected, a dialog box will appear allowing you to enter the name of the file (default file name = EZMICRO.PLT) to receive the graphical memory plot. Also, you must enter the starting and ending address range (of memory) that will be included in this memory plot.

Note: Before we begin our next example, first select the Load S-Record command from the FILE menu and select the file "plot1.s19". This will ensure that you get the same results as shown in this tutorial.

For this example, select the Memory Dump command from the FILE menu and the following dialog box will appear.

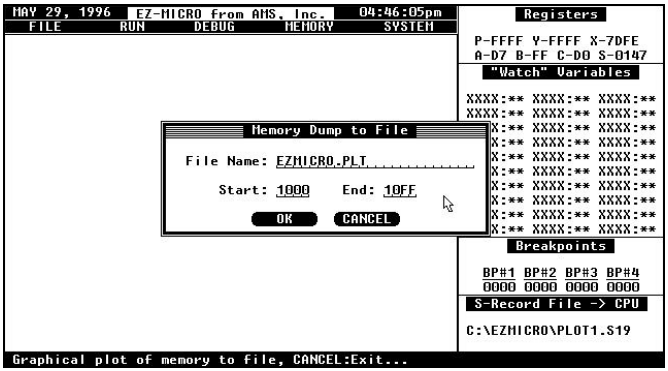
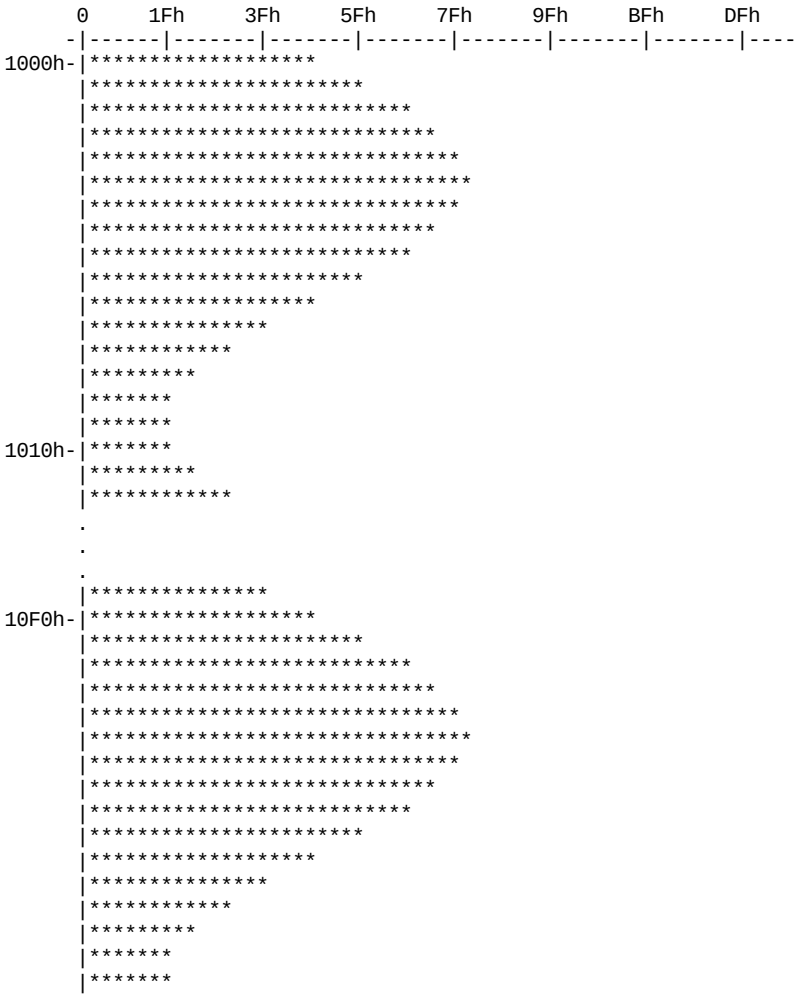


Figure 11: Memory Plot to a File.

The file name "ezmicro.plt" will do for this example. Enter "1000" for the starting address and enter "10FF" for the ending address. Press the OK button to accept the parameters entered in the dialog box (Figure 12).

We can use the EZ-MICRO Manager's editor to view the results of the memory plot that was just generated. Select the Edit Source command from the FILE menu. The name of the last file that was edited, "ezmicro.hex", will be displayed in the dialog box that appears. Press the ESC key (or the right mouse button) to remove this file name and enter "ezmicro.plt". Now press the ENTER key (or click the left mouse button) and the following memory plot should be displayed. When you have finished viewing the memory plot, press the ESC key to exit the editor.

EZ-MICRO "Graphical" plot of memory starting @ 1000h...



Example 3: EZMICRO.PLT (Memory Plot).

Save WaThe "Save Watches" command in the FILE menu is used to save the current watch variable addresses that have been entered into the watch variable's address table. Note: Since there are no watch variables currently entered in the table, this command will appear "*grayed out*". A watch variable is simply a memory location on the EZ-MICRO CPU-11 that you wish to monitor the contents of.

The EZ-MICRO Manager software allows the user to enter up to thirty addresses (or memory locations) into the watch variable's address table. The Save Watches command is used to save this list of addresses so that they can be quickly re-loaded when the EZ-MICRO Manager software is re-started. This saves the user from having to re-enter the list each time the EZ-MICRO Manager software is run.

To demonstrate this feature, we must first enter some hex addresses into the watch variable's address table. To do this, select Add Variable from the DEBUG menu and you will be prompted to enter a hex address. Enter the following hex addresses: 1000, 1001, 1002, 1003, 1004, 1005. After you have entered the last hex address, press the ESC key (or the right mouse button) to exit the Add Variable command. Upon pressing the ESC key, you should notice that the Watch Variables window now displays the six addresses (along with the contents of each memory location) that were just entered.

Next, select the Save Watches command from the FILE menu (it should no longer be "grayed out") and you will be prompted to enter the name of the file to store the watch variable addresses. Since we have not previously saved or loaded a watch variables file, the file name "noname.wch" will appear in the dialog box. Press the ESC key (or the right mouse button) to remove the existing name in the dialog box and enter "test.wch" and press the ENTER key (or the left mouse button) to save the current watch variables.

Load Watches

The "Load Watches" command in the FILE menu is used to load the addresses of watch variables that were previously saved using the Save Watches command. Note: When the Load Watches command is selected, any watch variables that are currently defined in the watch variable's address table will be deleted.

To demonstrate this feature, we will first remove all of the watch variables that were entered in the last example. To do this, select Clear All (just below the Remove Variable command) in the DEBUG menu. This will clear all of the addresses (displayed in the Watch Variables window) from the watch variable's address table. Next, select Load Watches from the FILE menu and you will be prompted to enter the name of the watch variables file to be loaded. The file name that is currently displayed, "test.wch", will do for this example, therefore press the ENTER key (or the left mouse button) to accept this file name. The watch variable addresses (1000, 1001, 1002, 1003, 1004 and 1005) should now be displayed in the Watch Variables window.

DOS Shell

Use this command to temporarily shell to DOS. While in the DOS shell, you may execute any of the DOS commands. However, depending on the amount of memory that is available, you may not be able to run or execute other programs while in this DOS shell. You will "shell" out into the same directory that the EZ-MICRO Manager software is running. The prompt that is displayed will not show you the current directory that you are in. Instead, the following prompt will appear:

DOS>

To return to the EZ-MICRO Manager, simply ENTER a "blank line" at the prompt.

Exit

In order to "exit" the EZ-MICRO Manager software normally, select this command. After selecting this command, you will return to the DOS prompt.

Note: If you are evaluating the EZ-MICRO CPU-11 microprocessor board and you are running the EZ-MICRO Manager software in the "normal" mode, then any changes that have been made to the host serial port (i.e. baud rate, parity, word length and stop bits) using the Configure Port option will be saved to the configuration file (ezmicro.cfg). These serial port parameters will be used again the next time the program is started.

The Run Menu

The RUN menu contains commands that are used when executing user code on the EZ-MICRO CPU-11 microprocessor board. Note: All of the commands in the RUN menu will be "*grayed out*" when running the EZ-MICRO Manager software in the demo mode. The EZ-MICRO CPU-11 hardware must be present in order to select any of the commands in the following section.

Go

The "Go" command in the RUN menu allows the user to initiate user program execution (free run in real time). The user may optionally specify a starting address where execution is to begin. Otherwise, execution begins at the current program counter (or P-register) address. Program execution continues until a breakpoint is encountered or until the EZ-MICRO CPU-11 master reset button (S1) is pressed. If a valid breakpoint is not reached by the executed program, then you will need to press the master reset button in order to stop program execution.

Note: The contents of the Registers and Watch Variables windows are only updated after program execution has stopped. Pressing the EZ-MICRO CPU-11 master reset button will remove the breakpoints that have been entered into the breakpoint address table.

Example: If you are evaluating the EZ-MICRO CPU-11 microprocessor board and Keypad/Display Lab board, then you will want to first assemble the source file "keydemo.asm" by selecting the Assemble Source command from the FILE menu. After the source file has been assembled, select Load S-Record from the FILE menu and select the assembled S-Record file "keydemo.s19" to be downloaded to the EZ-MICRO CPU-11 microprocessor board.

Next, select Go from the RUN menu and enter "1000" as the address of the first instruction to be executed. This will begin the execution of the program that we have downloaded to the EZ-MICRO CPU-11. The "keydemo" program will scan the keypad (on the Keypad/Display Lab board) for user input. Pressing a key on the keypad will display the corresponding number on all three LED displays and also echo the key pressed to the screen of the host computer or terminal.

To stop the program execution, press the master reset button (S1) on the EZ-MICRO CPU-11 microprocessor board. The EZ-MICRO Manager software will sense that a reset has occurred and will automatically exit from the Go command. Note: If the EZ-MICRO Manager does not automatically exit, then simply press the ESC key after pressing the master reset button.

To demonstrate the ability of the EZ-MICRO CPU-11 to stop program execution at a breakpointed address, select the Set Breakpoint command from the DEBUG menu and enter the breakpoint address "10A0". This address corresponds to an instruction (in the program that was downloaded to the EZ-MICRO CPU-11) that senses when the number "0" is pressed on the keypad. Select Go once more from the RUN menu and enter "1000" as the address of the first instruction to be executed. Press all of the numbers on the keypad (except for "0"). Notice that everything is functioning as before. Now press the "0" key and the EZ-MICRO Manager software will sense that a breakpoint has been encountered and program execution will be stopped.

Call

The "Call" command in the RUN menu allows the user to execute a user program subroutine. Execution starts at the current program counter (or P-register) address, unless a starting address is specified. Program execution continues until a breakpoint is encountered or until the EZ-MICRO CPU-11 master reset button (S1) is pressed. If a valid breakpoint is not reached by the executed program, then you will need to press the master reset button in order to stop program execution.

Note: The contents of the Registers and Watch Variables windows are only updated after program execution has stopped. Pressing the EZ-MICRO CPU-11 master reset button will remove the breakpoints that have been entered into the breakpoint address table.

Stop At

The "Stop At" command in the RUN menu causes the user program to be executed one instruction at a time until the specified address is encountered. Execution begins with the current program counter (or P-register) address and stops just before execution of the instruction at the specified stop address. This command should only be used when the contents of the program counter is known. The Stop At address that is specified may be RAM or ROM memory and must be the address of an opcode. If the Stop At address is not the address of an opcode, then you will need to press the master reset button in order to stop program execution.

Note: The contents of the Registers and Watch Variables windows are only updated after program execution has stopped. Pressing the EZ-MICRO CPU-11 master reset button will remove the breakpoints that have been entered into the breakpoint address table.

The EZ-MICRO Manager software will sense when the "stop at" address has been reached and will automatically exit from the Stop At command. Note: If the EZ-MICRO Manager does not automatically exit and a valid "stop at" address has been reached, then simply press the ESC key to exit the Stop At command.

Proceed

The "Proceed" command in the RUN menu is used to proceed or continue program execution without having to remove assigned breakpoints. This command is used to bypass assigned breakpoints in a program executed by the GO command. The program will continue to "proceed" to the next assigned breakpoint. If a valid breakpoint is not reached by the executed program, then

you will need to press the master reset button in order to stop program execution.

Note: The contents of the Registers and Watch Variables windows are only updated after program execution has stopped. Pressing the EZ-MICRO CPU-11 master reset button will remove the breakpoints that have been entered into the breakpoint address table.

The EZ-MICRO Manager software will sense when the next valid breakpoint is reached and will automatically exit from the Proceed command. Note: If the EZ-MICRO Manager does not automatically exit and a valid breakpoint has been reached, then simply press the ESC key to exit the Proceed command.

Trace

The "Trace" command in the RUN menu allows the user to monitor program execution on an instruction-by-instruction basis. The user may optionally execute several instructions at a time by entering a count value (up to $FF). Execution starts at the current program counter (or P-register) address. Note: Jumper location H11 on the EZ-MICRO CPU-11 microprocessor board must have a jumper strap inserted for this command to function properly.

The current register contents will be displayed along with each instruction that is executed in the Main Display window. Press the PgDn and PgUp keys to scroll the trace information forward and backward (or point at the right and left arrow buttons with the mouse and click the left mouse button). Pressing the "S" key will allow the user to save the trace information to a DOS file. When you have finished viewing the trace information, press the ESC key (or the right mouse button).

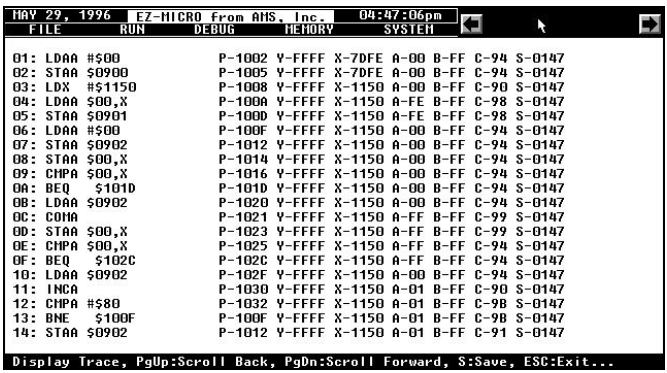


Figure 12: Trace Program Execution.

Step

The "Step" command in the RUN menu is used to execute each instruction one at a time. Note: Jumper location H11 on the EZ-MICRO CPU-11 microprocessor board must have a jumper strap inserted for this command to function properly. Execution starts at the current program counter (or P-register) address. Each instruction that is executed will be displayed in the Main Display window. To execute the next instruction, simply press the PgDn key on the keyboard (or point at the down arrow button with the mouse and click the left mouse button). The register contents in the Registers display window and the Watch Variables will be updated after each instruction that is executed. To exit from this command, press the ESC key (or the right mouse button).

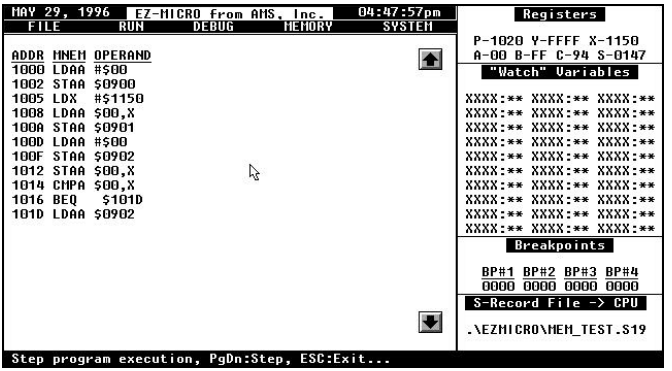


Figure 13: Step Program Execution.

Reset

The "Reset" command in the RUN menu is used to initiate a hardware reset on the EZ-MICRO CPU-11 microprocessor board. If you find that the EZ-MICRO Manager software is no longer communicating properly with the EZ-MICRO CPU-11, then select this command. After selecting this command, press the EZ-MICRO CPU-11 master reset button (S1). A message should be displayed stating that a "hardware reset has been detected".

Pressing the master reset button on the EZ-MICRO CPU-11 will "clear" the current breakpoints that have been set. However, any source code that has been downloaded into user memory will not be affected by this hardware reset.

The Debug Menu

The DEBUG menu contains commands that are used for debugging/executing user code downloaded to the EZ-MICRO CPU-11 microprocessor board.

Modify Registers

The "Modify Registers" command in the DEBUG menu is used to modify the EZ-MICRO CPU-11 program counter (P), Y-index (Y), X-index (X), A-accumulator (A), B-accumulator (B), Condition Code (C) and Stack Pointer (S) register contents.

Note: This command can also be selected by pointing at the Registers window (in the upper right corner of the screen) with the mouse and clicking the left mouse button.

When this command is selected, a dialog box will appear displaying the current register contents. Use the up or down keyboard arrow keys to select (or highlight) one of the registers that you want to modify. Once the desired register is selected, you may begin typing to enter the new hexadecimal value for the register. If you are using a mouse, simply point at the register that you want to modify and click the left mouse button.

When you have finished modifying the registers, press the HOME key to select the OK button and then press ENTER. If you are using a mouse, point at the OK button and click the left mouse button. Once the OK button is selected, the register contents (displayed on the screen and on the EZ-MICRO CPU-11 microprocessor board) will be updated.

If you do not want to save the changes that were made to the registers, then press the END key to select the CANCEL button and then press ENTER to discard the changes. If you are using a mouse, point at the CANCEL button and click the left mouse button.

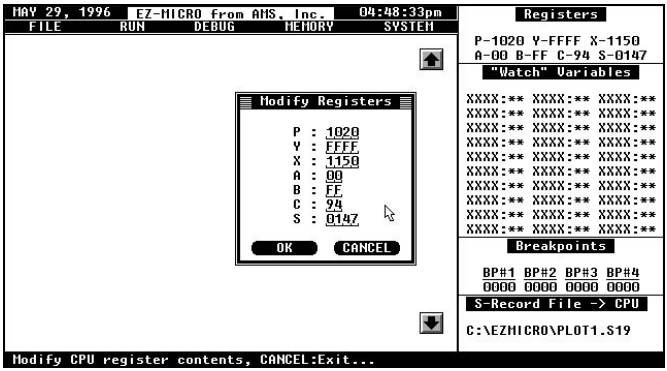


Figure 14: Modify Registers.

Add Variable

The "Add Variable" command in the DEBUG menu is used to add watch variable addresses into the watch variable's address table. A watch variable is simply a memory location on the EZ-MICRO CPU-11 microprocessor board that you wish to monitor the contents of.

The EZ-MICRO Manager software allows the user to enter up to thirty addresses (or memory locations) into the watch variable's address table. The Save Watches command (in the FILE menu) can be used to save this list of addresses so that they can be quickly re-loaded when the EZ-MICRO Manager software is re-started. This saves the user from having to re-enter the list each time the EZ-MICRO Manager software is run.

To demonstrate this feature, select the Add Variable command from the DEBUG menu and you will be prompted to enter a hex address. Enter the following hex addresses: 100A, 100B, 100C, 100D, 100E and 100F. After you have entered the last hex address, press the ESC key (or the right mouse button) to exit the Add Variable command. Upon pressing the ESC key, you should notice that the Watch Variables window now displays the six addresses (along with the contents of each memory location) that were just entered.

Remove Variable

The "Remove Variable" command in the DEBUG menu is used to remove the address of a specific watch variable from the watch variable's address table. When prompted, enter the hex address of the watch variable that you want to remove. Note: This command will be "grayed out" when there are no watch variables currently defined in the watch variable's address table:

Clear All Variables

The first "Clear All" command in the DEBUG menu is used to remove all of the watch variable addresses that have been entered into the watch variable's address table. Note: This command will be "grayed out" when there are no watch variables currently defined in the watch variable's address table:

Set Breakpoint

The "Set Breakpoint" command in the DEBUG menu is used to place an address into the breakpoint address table. When prompted, enter the hexadecimal address of where the breakpoint is to be inserted. A maximum of four breakpoints may be set and will be displayed in the Breakpoints window. If four breakpoints have already been set, the message "Breakpoint table is FULL" will appear when a fifth breakpoint is entered.

Whenever the Go, Call or Proceed commands are selected (from the RUN menu), then "breakpoints" are inserted into the user code at the address specified in the breakpoint address table. During user program execution, the program will stop execution immediately preceding the execution of any instruction's address that is in the breakpoint address table.

Breakpoints are implemented by placing a software interrupt (SWI) at each address specified in the breakpoint address table. The SWI service routine saves and displays the internal machine state, then restores the original opcodes at the breakpoint location before returning control back to the EZ-MICRO Manager program.

Note: SWI opcodes cannot be executed or breakpointed in user code because the BUFFALO monitor program uses the SWI vector. Only RAM locations can be breakpointed. Branch on self instructions cannot be breakpointed.

Remove Breakpoint

The "Remove Breakpoint" command in the DEBUG menu is used to remove one of the breakpoints that have been set in the breakpoint address table. When prompted, enter the address of the breakpoint that is to be removed from this list.

Clear All Breakpoints

The second "Clear All" command in the DEBUG menu is used to remove all of the breakpoints that have been set in the breakpoint address table.

Line Assemble

The "Line Assemble" command in the DEBUG menu is a single line assembler/disassembler that can be used to input code directly into user RAM. Note: This command will be "grayed out" when running the EZ-MICRO Manager software in the demo mode.

When the Line Assemble command is selected, you will be prompted to enter the starting address to begin the "in-line" assembly. Next, you will be prompted to enter a source line. Each source line is converted into the proper machine language code and is stored in memory overwriting previous data on a line-by-line basis at the time of entry.

All valid opcodes are converted to assembly language mnemonics. All invalid opcodes will be displayed as an error message. If an error does occur, then the same address location is re-opened and the source line can be re-entered.

Note: Pressing the ENTER key (without entering anything) will leave the current instruction unchanged and advance to the next valid instruction.

All numerical values are assumed to be hexadecimal. Therefore, no base designators (such as $, %, etc.) are allowed. Operands must be separated by one or more spaces. Any characters after a valid mnemonic are ignored. Addressing modes are designated as follows:

a.) Immediate addressing is designated by preceding the address with a # sign.

b.) Indexed addressing is designated by a comma. The comma must be preceded by a one byte relative offset (even if the offset is 0) and must be followed by an X or Y designation of which index register to use.

c.) Direct and extended addressing is specified by the length of the address operand (1 or 2 digits specifies direct, 3 or 4 digits specifies extended).

d.) Relative offsets for branch instructions are computed by the assembler. Therefore, the valid operand for any branch instruction is the branch-if-true address, not the relative offset.

The Memory Menu

The MEMORY menu contains commands that are used for modifying/displaying user memory located on the EZ-MICRO CPU-11 microprocessor board.

Memory Dump

The "Memory Dump" command in the MEMORY menu is used to display the hexadecimal contents of user memory (located on the EZ-MICRO CPU-11 microprocessor board) to the screen.

For example, select the Memory Dump command from the MEMORY menu and you will be prompted to enter the starting memory location to begin the hexadecimal memory dump. Enter the hex address "1000" for this example and press the ENTER key (or click the left mouse button) to accept this hex number. Note: The starting address that is specified may be either RAM or ROM address space (0000h-FFFFh).

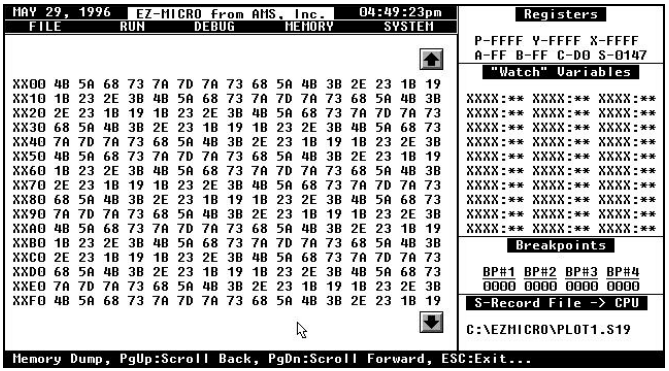


Figure 15: Hexadecimal Memory Dump.

Note: The memory addresses will appear as "XX00" only during the demo mode. This is done since there are only 256-bytes of memory (0000h-00FFh) available when running the EZ-MICRO Manager in the demo mode.

Use the PgUp and PgDn keys to view the previous or next page of the hexadecimal memory dump (or point at the up and down arrow buttons with the mouse and click the left mouse button). When you have finished viewing the hexadecimal memory dump, press the ESC key (or the right mouse button).

Disassemble

The "Disassemble" command in the MEMORY menu is used to display a disassembled listing of the program code to the screen. Note: This command will be "grayed out" when running the EZ-MICRO Manager software in the demo mode. When the Disassemble command is selected, you will be prompted to enter the address of the first line of code (or instruction) to be disassembled. Use the PgDn and PgUp keys to scroll the disassembly listing forward and backwards (or point at the down and up arrow buttons with the mouse and click the left mouse button). Note: You will not be able to scroll backwards past the first instruction that is disassembled. Press the ESC key (or the right mouse button) to exit the disassemble command.

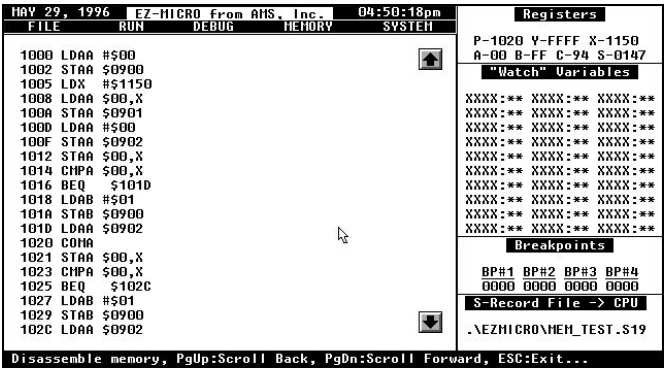


Figure 16: Disassembled Memory Dump.

Memory Plot

The "Memory Plot" command in the MEMORY menu allows the user to view the contents of memory "graphically" on the screen. This is especially useful when sampling an analog signal (using the ADC/DAC project lab board) and storing the resulting data into RAM.

To demonstrate this feature, first select the Load S-Record command from the FILE menu. When prompted, enter the file name "plot1.s19" and press ENTER (or the left mouse button) to load the S-Record file into memory.

Next, select the Memory Plot command from the MEMORY menu and you will be prompted to enter the starting address of memory to be plotted. Enter "1000" and press the ENTER key (or the left mouse button). The Main Display window will expand and a plot of the contents of memory will be displayed.

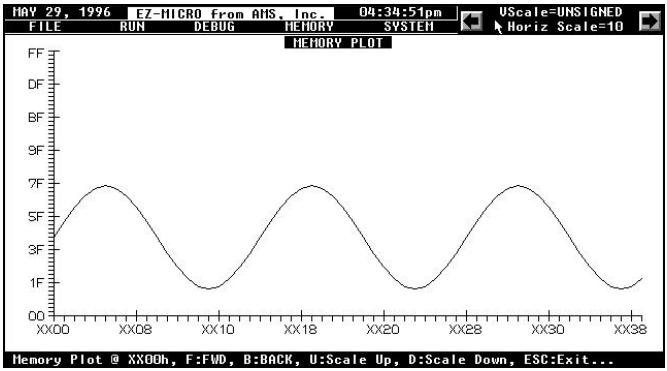


Figure 17: Memory Plot.

The vertical axis represents the amplitude (or hex value) of the byte of data. The vertical axis can be changed to plot signed or unsigned data by pressing the letter "V". The current vertical axis "type" will be displayed in the upper right corner of the screen.

The horizontal axis represents the memory address location, starting with the first address specified. The horizontal axis can be scaled UP (to display fewer bytes of data) or scaled DOWN (to display more bytes of data) by pressing the "U" and "D" keyboard keys respectively. The current horizontal axis "scale" will be displayed in the upper right corner of the screen.

Pressing the letter "B" and "F" on the keyboard (or pointing at the left and right arrow buttons with the mouse and clicking the left mouse button) will go backwards and forwards in memory.

Pressing the letter "A" on the keyboard will allow you to specify a new starting address to begin the memory plot.

When you have finished viewing the "graphical" plot of memory on the screen, press the ESC key (or the right mouse button) to exit the Memory Plot command.

Memory Modify

The "Memory Modify" command in the MEMORY menu allows the user to modify the contents in user memory beginning at the specified address.

To demonstrate this feature, select the Memory Modify command from the MEMORY menu. When prompted, enter "1000" as the starting hexadecimal address of memory to begin modifying. You will then be prompted to enter the

new contents for this memory location. Enter "00" when prompted. After entering "00", you will be prompted to enter the contents for the next memory location. This time enter "11". Continue to enter the following four hex values when prompted: "22", "33", "44" and "55".

Note: Pressing ENTER without entering a value will leave the current memory location unchanged and increment to the next memory location. To exit the Memory Modify command, simply press the ESC key (or the right mouse button).

To view the changes that have been made to the six memory locations (1000h-1005h), select the Memory Dump command from the MEMORY menu and enter "1000" as the starting hex address to begin the memory dump. The following figure shows this memory dump. Notice that the first six memory locations show the hex numbers that were just entered. When you have finished viewing the hexadecimal contents of memory, press the ESC key (or the right mouse button).

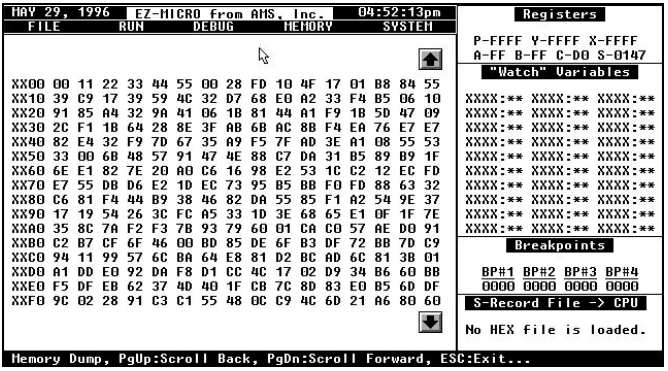


Figure 18: Memory Dump after Modifying Memory.

Block Fill

The "Block Fill" command in the MEMORY menu allows the user to repeat a specific pattern throughout a determined user memory range. To demonstrate this feature, select the Block Fill command from the MEMORY menu and the following dialog box will be displayed.

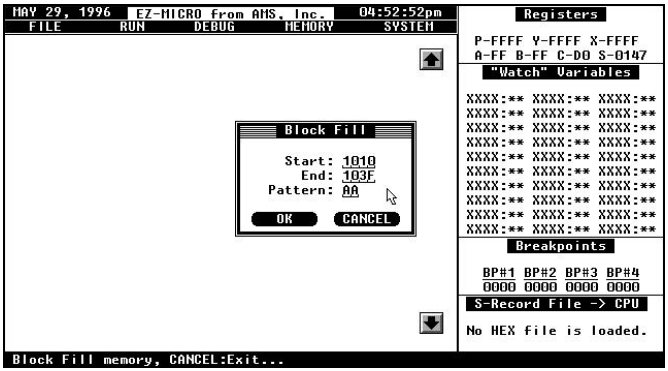


Figure 19: Block Fill Memory with Pattern.

Enter "1010" for the starting address location, enter "103F" for the ending address location and enter "AA" for the pattern. Press the OK button to accept the parameters you have just entered.

To view the changes that have been made to the memory locations (1010h-103Fh), select the Memory Dump command from the MEMORY menu and enter "1000" as the starting hex address to begin the memory dump. The following figure shows this memory dump. Notice that memory locations XX10h to XX3Fh are filled with the value AAh. When you have finished viewing the hexadecimal contents of memory, press the ESC key (or the right mouse button).

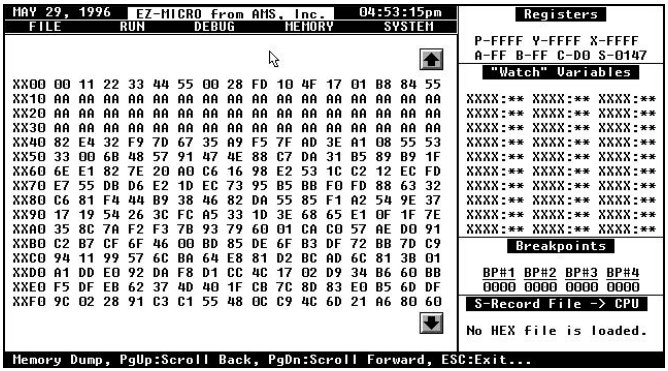


Figure 20: Memory Dump after Filling Block of Memory.

Block Move

The "Block Move" command in the MEMORY menu allows the user to copy or move a "block" of memory to a new memory location. To demonstrate this

feature, select the Block Move command from the MEMORY menu and the following dialog box will be displayed.

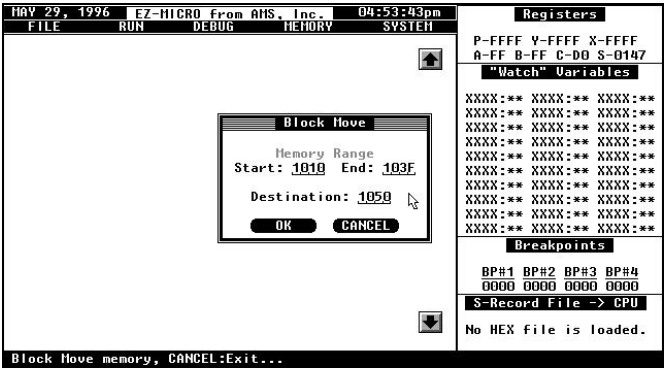


Figure 21: Block Move Memory.

Enter "1010" for the starting address location and enter "103F" for the ending address location of the block of memory to move. Enter "1050" for the destination hex address to receive the block (48-bytes) of memory. Press the OK button to accept the parameters you have just entered.

To verify that the block of memory (1000h-103Fh) was copied to 1050h, select the Memory Dump command from the MEMORY menu and enter "1000" as the starting hex address to begin the memory dump. The following figure shows this memory dump. Notice that memory locations XX50h to XX7Fh are filled with the value AAh. When you have finished viewing the hexadecimal contents of memory, press the ESC key (or the right mouse button).

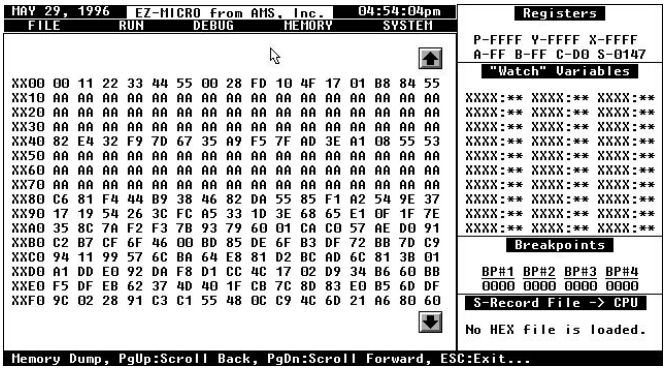


Figure 22: Memory Dump after Moving Block of Memory.

The System Menu

The SYSTEM menu contains commands that are used for configuring options that concern the operation of the EZ-MICRO Manager software.

Configure Port

The "Configure Port" command in the SYSTEM menu is used to configure the COM port (or serial port) of the host computer or terminal for communicating with the EZ-MICRO CPU-11 microprocessor board.

When the EZ-MICRO CPU-11 is first powered-on, its serial port will be configured to 9600 baud, no parity, eight bit word length and one stop bit. The options that you select for the host serial port must match these settings (9600,N,8,1) exactly or you will not be able to communicate with the EZ-MICRO CPU-11. Note: When the EZ-MICRO Manager software is started (in the "normal" mode) for the first time, the Configure Port dialog box will automatically be displayed so that you can select the COM port that you will be using to communicate to the EZ-MICRO CPU-11 microprocessor board.

When the "Configure Port" command is selected from the SYSTEM menu, a dialog box will appear allowing you to select the port address, baud rate, parity, word length and stop bit parameters. The options that are selected in the Configure Port dialog box will be saved to the EZ-MICRO Manager's configuration file (ezmicro.cfg) when you exit the program normally. That is, by selecting Exit from the FILE menu. Once the serial port options have been selected, the Configure Port dialog box will not be displayed again when the EZ-MICRO Manager software is re-started.

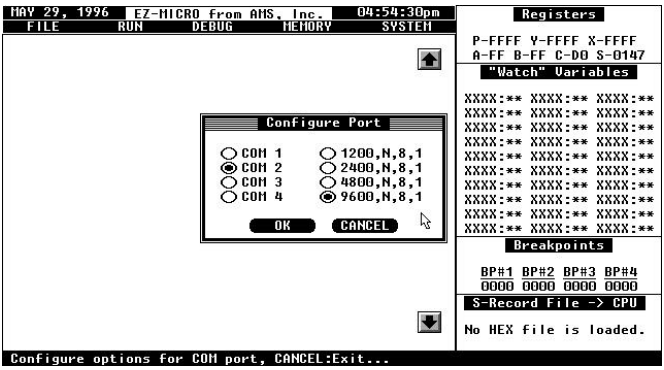


Figure 23: Configure Port Options.

It is possible to change the serial port parameters of the EZ-MICRO CPU-11's serial interface by configuring the serial port register ($002B). If changes are made to the serial port of the EZ-MICRO CPU-11, then you will need to select the identical parameters for the host computer's serial port. Note: If a reset occurs on the EZ-MICRO CPU-11, the serial interface (of the EZ-MICRO CPU-11) will return to its default value (9600,N,8,1).

Configure Colors

The "Configure Colors" command in the SYSTEM menu is used to configure the screen colors of the EZ-MICRO Manager's pull down menus, dialog boxes and display windows. Changing the screen colors may require that you exit the software and re-start it for the changes to take affect. The screen color selections are saved to the configuration file "ezmicro.clr".

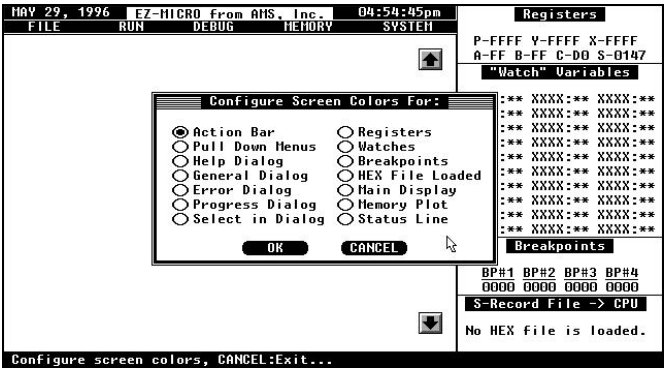


Figure 24: Configure Screen Colors.

Change DIRectory

The "Change DIR" command in the SYSTEM menu is used to change the default search path when the EZ-MICRO Manager is searching for files. Note: The default search path is the directory where the EZ-MICRO Manager program is currently running.

For example, select the Assemble File command from the FILE menu. Press the ESC key (or the right mouse button) to remove the existing file name in the dialog box. Next, press the ENTER key twice (or click the left mouse button twice) and a listing of the available assembly language source files (in the "c:\ezmicro" directory) will be displayed. The directory "c:\ezmicro" is the default search path.

If we wanted, we could move all of the assembly language source files (*.asm) into a directory called "c:\lessons". We would then use the Change DIR command to change the search path to "c:\lessons". Now when the Assemble File command is selected, the EZ-MICRO Manager will search the directory "c:\lessons" for any source files.

Scroll When Bump

A check-mark next to this option in the SYSTEM menu will allowthe user to scroll a Memory Dump display or a Disassembled memory dump by simply touching the top or bottom of the MAIN display window with the mouse. To enable or disable this feature, simply select this option in the menu. This will toggle the check-mark (on and off) for this option.

About EZ-MICRO

Selecting this command will display the current version of the EZ-MICRO Manager software that is currently running.

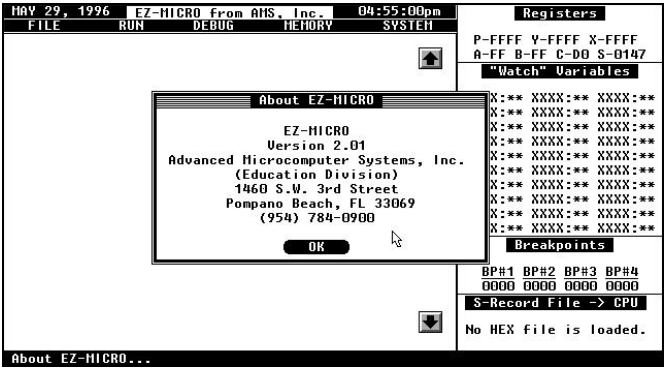
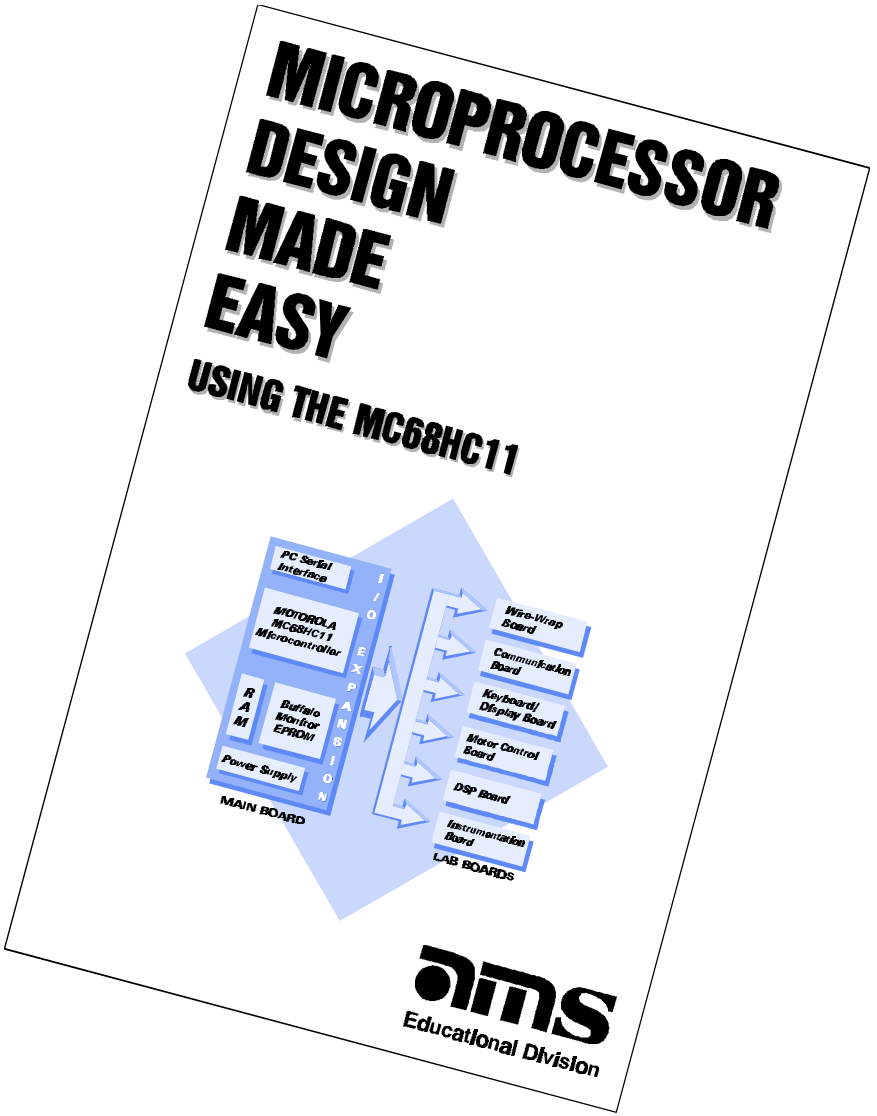


Figure 25: About EZ-MICRO

Features of the EZ-Micro Workbook



1 Introduction to Microprocessors

- The history and evolution of the microprocessor.
- The different types of microprocessor available and their features.
- The difference between 8-bit, 16 bit and 32-bit microprocessor.

2 The MC68HC11 Architecture & its addressing modes

After completing this chapter you will learn:

- The internal structure of the MC68HC11.
- The different type of the addressing modes.
- How binary information moves inside an MC68HC11 based system.

3 Programming the MC68HC11

After completing this chapter you will learn:

- The different instructions groups of the MC68HC11
- How to create common data structure in MC68HC11 assembly.
- How does EZ-MICRO TUTOR manager software works.
- The commands of the EZ-MICRO TUTOR Manager Software.
- How to use an assemble and debugger.

4 EZ-MICRO Tutor Manager

After completing this lesson, you will know:

- How the AMS Tutor Manager software works.
- Commands of the AMS Tutor Manager software.
- The feature and capabilities of the AMS Tutor Board.
- How to download programs to the AMS Tutor Board.

5 Software Interface to AMS Tutor Board

The objectives of this laboratory are:

- To become familiar with the AMS 68HC11 Tutor package
- To set up the PC/AMS TUTOR DEVELOPMENT system.
- To be introduced to the assembler source code syntax & assembly process.
- Exercise the software development tools

6 Assembling Language Programming

- This lab will involve developing code from a specification. The specification will be translated into pseudo-code, then into assembly language, then downloaded to Tutor board and debugged.

7 Application Programming: Look up Tables and Subroutines

- This laboratory will involve the exploration of data arrays and look-up tables (LUTs). These data handling techniques have many applications in embedded systems. They can provide the embedded systems designer the capability to generate or translate data that cannot be calculated due to timing constraints.
- Subroutines: Use of subroutines to make your program efficient structured program

8 How the AMS Tutor Board interfaces with outside world

- This laboratory will involve interfacing input and output devices to the AMS Tutor Board. The input device will be a keypad, and the output device will be three 7 segment LED display.
- Asynchronous Vs Synchronous send handshake
- Send/Receive handshake
- Interrupt Vs Polled
- Single direction Vs Bi-directional Vs Bus Interconnection

9 Concept of Interrupt for measuring period of input signal using Timer

- This laboratory experience will involve two major concepts: measuring the period of an input signal using the input capture timer, and interrupts. These two methods work together to perform a complex task with an efficient combination of hardware and software.

10 Generate output pulse using internal timer.

- The 68HC11 Microcontroller timer system is a flexible and powerful tool for solving a wide variety of problems that involve measuring and generating time series waveforms. This lab will involve using the timer features that output pulses at precise time intervals.

11 Introduction to Analog to Digital conversion.

- Analog to digital conversion is the process of transforming a continuous signal into a discrete representation. This process has been described as the process of converting real world signals to a form that is used in the artificial world of digital systems. Extensive theoretical and mathematical studies have been performed to analyze the A/D process. This introduction will try to identify the key concepts involved with A/D conversion.

12 Analog Input and Output

- In this laboratory, the program will be developed to develop the driver to test the analog input and output features.
- The specification is as follows: Write a program which will generate a 2 kHz sample rate clock to convert an analog sample and output this sample value to the DAC port. Set up a function generator to provide a sinusoidal signal with an amplitude range of 0 to 3 volts and a frequency range of 5 to 3 kHz. Use an oscilloscope to monitor the waveforms at various points in the circuitry to observe the action of the DSP board at different input frequencies.

13 Introduction to Digital Signal Processing

- Digital Signal Processing (DSP) describes the process of manipulating signals while they are represented in a discrete format. Computer systems are capable of performing a wide variety of processes on signals under software control. One common class of DSP algorithms is the digital filter.

Notes:

Notes:

Other AMS Educational Products

EZ-ROUTE Tutor™ Electronic CAD Courseware

Evaluation Package Includes:

- EZ-ROUTE™ Std Software Package
  (Schematic Capture, Symbol Editor, PCB Layout)
- 300+ Page Student Courseware Book
- Instructor's Lab Manual

EZ-ROUTE Tutor™ Advanced Electronic CAD Courseware

Evaluation Package Includes:

- EZ-ROUTE™ Pro Software Package
  (Schematic Capture, PCB Layout and Auto Router)
- 300+ Page Student Courseware Book
- Instructor's Lab Manual

EZ-MICRO Tutor™ Microprocessor Design Course

Evaluation Package Includes:

- EZ-MICRO™ Lab Software Package
- 68HC11 CPU Board
- Lab Project Board
- 300+ Page Student Courseware Book
- Instructor's Lab Manual

EZ-DSP Tutor™ Digital Signal Processing Course

Evaluation Package Includes:

- EZ-DSP™ Lab Software Package
- EZ-DSP™ CPU Board
- Lab Project Board
- 300+ Page Student Courseware Book
- Instructor's Lab Manual

Note: **For pricing information please call 1-800-972-3733**

Notes: