VIII. ROM EMULATOR POD (RP-016*)

* On early versions of the ROM Emulator Pod, the catalog number on the Pod label is "RE-016" instead of "RP-016".

WARNING

BEFORE PLUGGING IN THE ML4400 TO AN AC POWER SOURCE, VERIFY THAT THE CORRECT VOLTAGE (115 OR 230) HAS BEEN SELECTED VIA THE VOLTAGE SELECTOR SWITCH AT THE UPPER RIGHT OF THE REAR PANEL.

A. GENERAL

The optional ROM Emulator Pod for the ML4400 is identical to the ROM Emulator Pod sold with Arium's earlier ML4100 logic analyzer. The ML4400 will accommodate one or two ROM Emulator Pods. When used, the second Pod is daisy-chained to the first Pod, which is the one closest to the ML4400.

The ROM Emulator Pod enables the ML4400 to emulate one or more 27XX-series EPROMS, giving it a powerful tool for debugging code in microproccesor-based systems. Code can be downloaded to the Emulator and then patched as required, via the ROM Emulator Editor screen. With one Pod, up to 16 Kbytes of memory may be emulated, and configured as four 2716s, four 2732s, two 2764s, or one 27128. With a second Pod connected to the first Pod, twice the memory can be emulated.

Four screens are used for setting up and operating the ROM Emulator, all of which are accessed via the Special Functions screen (S. FUNC key): RS232C Settings, ROM Emulator Configuration, ROM Emulator Transfers, and ROM Emulator Editor. The first two screens are used for setup; the latter two, for operation.

To specify or select parameters on any of these screens, cursor to a field, then use the keypad and the six softkeys to either select one of the given parameters (as indicated by the screen prompting labels), or enter the desired value directly into the field.

B. SETUP AND CONFIGURATIONS

Before transferring or editing data with the ROM Emulator Pod, the user must first specify the configurations of the ROM Emulator and of the ML4400's RS232C port, as described in Sections 1 and 2, below. Without such configuration, the ROM Emulator will function incompletely or cause errors.

1. CONFIGURATION OF RS232C SERIAL PORT

The ML4400's RS232C Serial Port is configured via the RS232C Settings Setup screen, which is accessed by depressing S. FUNC, 3. On this screen, the user can specify the following parameters:

- Baud rate (selected from 12 given rates) at which messages will be sent and received by the ML4400;
- Parity (on, off, or even);
- Character length (7 or 8 bits);
- How long the port should delay after transmitting a character;
- How long the port should delay after a carriage return (for devices which are slow to receive characters);
- What the End-of-File character should be;
- What the End-of-Line character(s) should be;
- Whether it should recognize XON & XOFF; and
- Whether it should recognize CLEAR TO SEND ("CTS").

For further details regarding the RS232C interface, see Section III-F.6, "RS-232C Port Settings".

2. CONFIGURATION OF ROM EMULATOR

The ROM Emulator Configuration screen is used to configure the way in which the ROM Emulator appears to the hardware, as well as to the user, when editing or downloading data. This screen is accessed via S. FUNC, 5. (This screen uses no softkeys.)

The user may specify the following parameters:

Whether two pods (if two are in use) are part of the same memory;
The address numeric base for editing (octal or hexadecimal);
Address and data widths in bits (address widths may be from 11 to 24 bits, and data width may be 8, 16, or 32 bits);
Device type for a given ROM Emulator Pod (2716, 2732, 2764, or 27128); and
Base addresses for the ROMs being emulated (the address in the target system where these ROMs will be chip-selected).

These parameters are selectable for each of the two ROM Emulator Pods, if two Pods are installed. When a second Pod is connected, the Configuration screen displays a second column of parameter selection fields.

For 32-bit-wide data, both Pods are necessary to provide 32 bits of data, and both must be part of the same Memory. Also, for a 27128, the 32-bit width is not available, and 16-bit width requires two ROM Emulator Pods.

The user may set base addresses for the ROMs being emulated, making editing and downloading convenient. For example, when the ROM Emulator is emulating four 2716s, one ROM may have a base address of 0, the next may have a base address of 800 hex, and the next, a base address of 1000 hex, etc.

The first field, "Pods part of same memory", is displayed only when two Pods are being used. When the Pods share the same memory, then the "Address base" and "Address width" fields are not displayed for Pod 2, since their values are the same as those for Pod 1.

To select the address base, enter C (octal) or H (hexadecimal) in the "Address base" field. (The selected base will also be used in the ROM Emulator Transfer and Editor screens.)

To specify the address width, enter the width (in bits) of the target processors's address bus (11-24) in the "Address width" field. (The specified width will also be used in the ROM Emulator Transfer and Editor screens.)

To specify the data width, enter the width (in bits) of the target processor's data bus (8, 16, or 32) in the "Data width" field. Widths over 32 are not allowed, and nonstandard widths of less than 32 will be rounded up to the next standard width. (The specified width will also be used in the ROM Emulator Editor screen.)

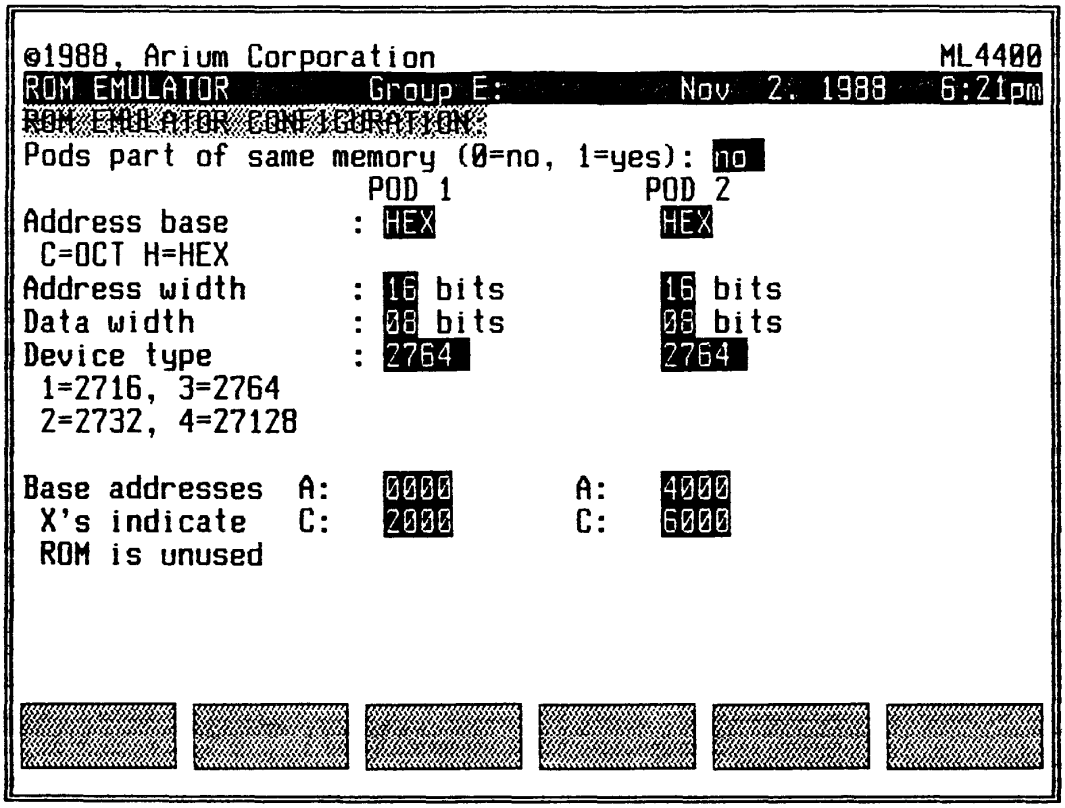
To select the ROM to be emulated, enter 1 (2716), 2 (2732), 3 (2764) or 4 (27128) in the "Device type" field.

To specify base addresses, use the keypad and ARROW keys to enter a value in each "Base addresses" field. To disable a ROM, enter Xs (Don't Cares) in the field.

Figure VIII-A

ROM EMULATOR CONFIGURATION SETUP SCREEN

34 lines



C. OPERATION

1. TRANSFERING DATA TO AND FROM THE ROM EMULATOR

Data is downloaded to the ROM Emulator Pod(s) and uploaded from the Pod(s) via the ROM Emulator Transfers screen, accessed via S. FUNC, 4. The available softkeys are UPLOAD (F3), DOWNLOAD (F4), and VERIFY (F5).

Before transferring data, the user must first configure the Emulator (via the ROM Emulator Configuration screen) and the RS232C serial port (via the RS232C Port Settings screen); see Section VIII-B, above.

There are two handshake methods of starting and stopping the RS232C data transfer between the ML4400 and the ROM Emulator Pod, as selected in the RS232C Port Settings screen. One uses the RTS and CTS lines of RS232C, and the other uses DC1 and DC3 (XON and XOFF) lines of U. S. ASCII.

If a parity, overrun or framing error occurs during a data transfer, the transfer will be aborted and an error message displayed on the Header line.

The ROM Emulator Transfers screen allows the user to specify the following parameters:

Which ROM Emulator Pod to download to or upload from;
Transfer range (lowest/highest addresses to be transferred to/from;
Transfer format for uploading (choice of 6 formats); and
Transfer type (upload, download, or verify).

To select the ROM Emulator Pod to use, enter 1 or 2 in the "ROM Emulator Pod" field. (If only one Pod is being used, or if both Pods are part of the same memory, only "1" may be selected.)

To specify the data transfer range, enter the beginning and ending addresses of the range in the "Low address" and "High address" fields. (When downloading, any data falling outside of this transfer range is ignored.)

To select the data transfer format for uploading, enter the number (1-6) designating the desired format in the "Transfer format" field. (The designations are: 8-bit Intel, 1; 16-bit Intel, 2; Motorola, 3; Tektronix hex, 4; extended Tektronix hex, 5; MOS Technology, 6.) This field determines byte ordering in Memory; the Intel formats cause data to be placed with the least significant bit first in Memory, while the other formats place the most significant bit first. The Motorola format supports S1, S2, and S3 records (16-, 24-, and 32-bit addresses, respectively).

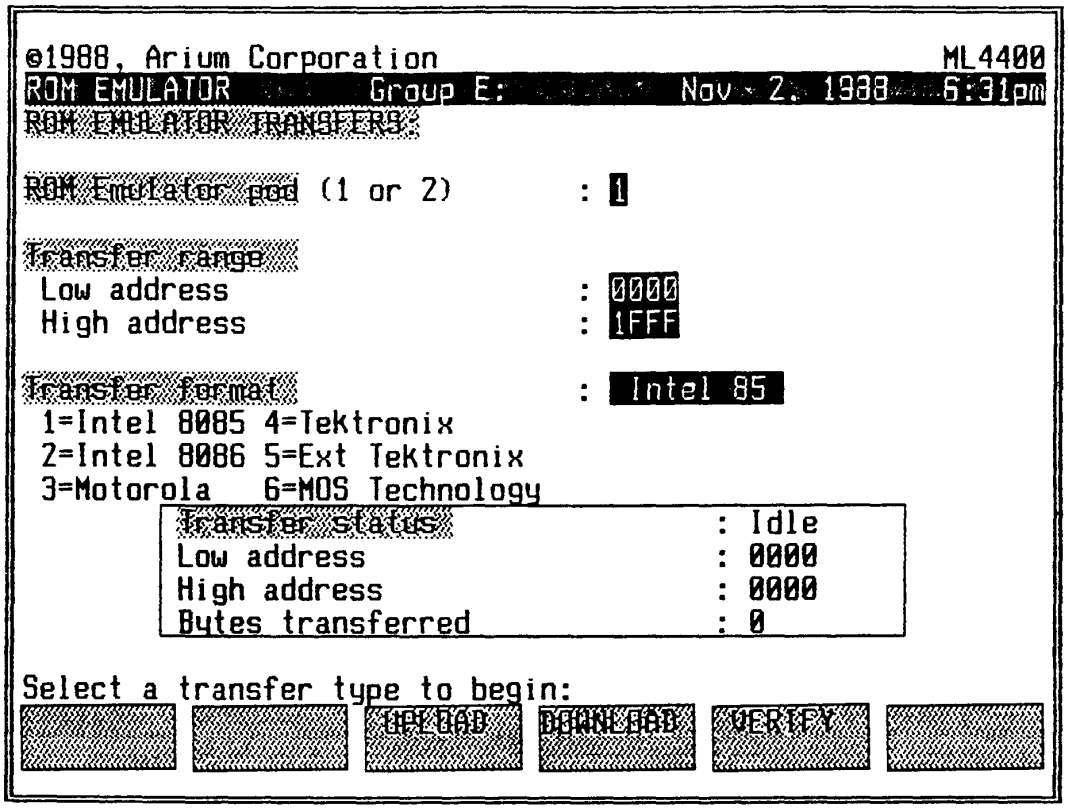
(When downloading, the ML4400 recognizes the record type and accepts it regardless of the value of the "Transfer format" field.)

To transfer data to or from the ROM Emulator, depress the DOWNLOAD or UPLOAD softkey twice. (The screen will prompt the user after the first key depression.) The "Transfer status" field is updated to show the operation selected. During the transfer, the number of records transferred is shown in the Header line. After the data transfer, the "Low address", "High address" and "Bytes transferred" fields are automatically updated.

To verify downloaded data (compare it to the contents of the ROM Emulator), depress VERIFY. To abort a data transfer, depress any key. (To leave the ROM Emulator Transfer screen, the transfer status must be "idle".)

Figure VIII-B

ROM EMULATOR TRANSFERS SCREEN



2. EDITING DATA IN THE EMULATED ROM

Data in the emulated ROMs may be edited via the ROM Emulator Editor screen, accessed via S. FUNC, 6. The available softkeys are CHANGE POD (F1), CHANGE BASE (F2), SUM MEMORY (F4), and FILL MEMORY (F5); the F3 and F6 softkeys are unused.

Before editing data, the user must first configure the ROM Emulator Pod (via the ROM Emulator Configuration screen).

The ROM Emulator Editor screen displays 16 rows of data with addresses on the left, and up to 16 columns of data on the right. (The actual amount of data depends upon the data width, data display base, address width, and address display base, as previously specified by the user in the ROM Emulator Configuration screen.)

Data is edited by using the ARROW keys for cursoring, and the keypad to enter new values in the data fields. The new value will be written to the Pod as soon as the cursor exits the field. While cursoring, the current address is highlighted at the far left of the current line. The least-significant digit of address is displayed on the third line.

During editing, the ROMs are not available to the target system; after editing, exit the Editor screen and reset the target processor.

The ROM Emulator Pod hardware can detect whether a ROM socket has been plugged in backwards, has not been plugged in, or has been plugged into a system with no power. If any of these conditions exists, a warning will be displayed when the ROM Editor screen is exited.

To access the other ROM Pod when using two Pods, depress the CHANGE POD softkey.

To access a particular address in Memory, use one of three methods:

Use the UP/DOWN ARROWS to move to the top of bottom of a screen, then to scroll one line at a time to the address.

Use the UP/DOWN ARROWS with the SHIFT key to move one screen at a time.

Cursor to any Address field, enter the desired address, and exit the field; the screen display will be redrawn, with the new address appearing at the top left.

To edit a Data field, cursor to the desired field and enter the new value via the keypad. The new data will be written to the Pod when the cursor exits the field.

To clear a field to zeroes, depress CLEAR; to undo the last clear, depress CLEAR again. To restore a field to its original value (before any editing), depress 2ND, CLEAR.

To select the number base for displayed data, use the CHANGE BASE softkey to roll between the five choices (ASCII, binary, octal, decimal and hexadecimal). (The address number base, as set in the (ROM Emulator?) Configuration screen, is unaffected.)

To cause a checksum to be computed, depress the SUM MEMORY softkey, which opens the Sum Memory window, then specify the range of Memory to sum by entering its low and high addresses, and depress SUM MEMORY again. The 32-bit sum will be displayed in the selected data display base. (For large ranges, the sum operation may take some time.)

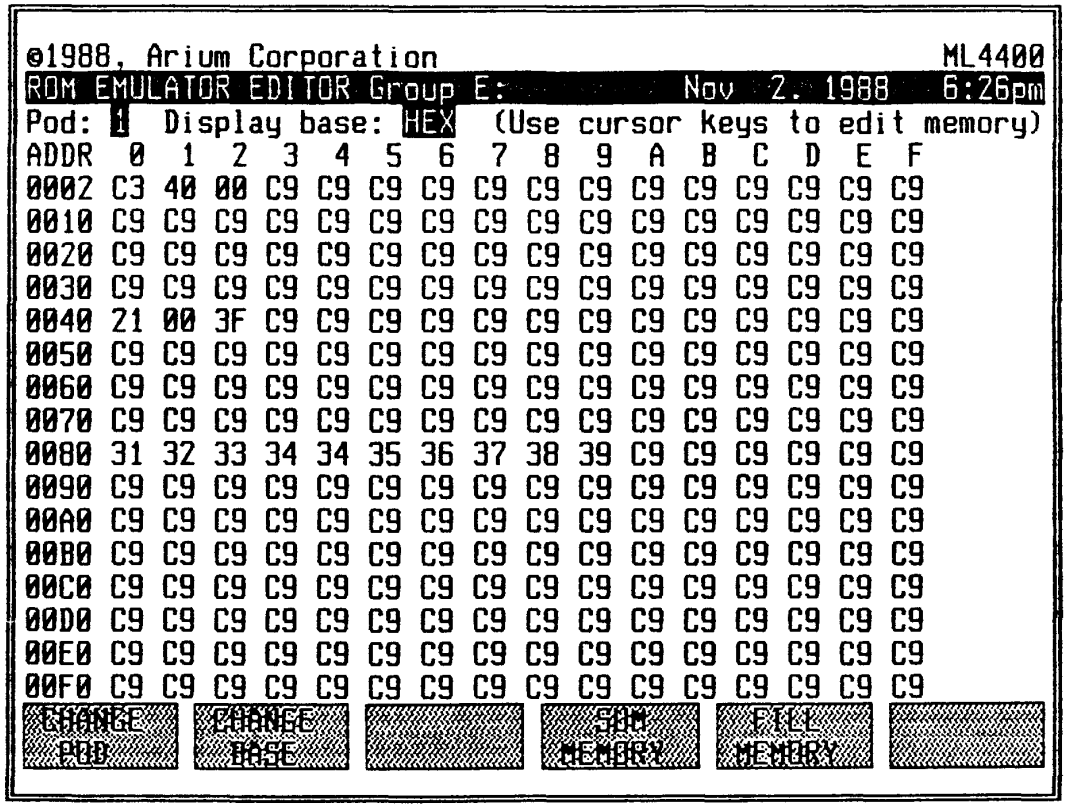
If the current data width is 8 bits, then all bytes in the range are summed; if the width is 16 bits, then every second byte is summed; and if the width is 32 bits, then every fourth byte is summed. This feature allows a particular ROM to be checksummed. For example: To checksum every byte in a 16-bit wide range, perform two sums, with the first at the starting address and the second at the starting address plus 1; then add the two sums together.

To close the Sum Memory window, depress any kwy.

To fill memory with desired constant data, depress the FILL MEMORY softkey, which opens the Fill window. Then specify the fill range by entering the low and high addresses, enter the fill data, and depress FILL MEMORY again. The Fill window will close and the data will be redisplayed. (For large ranges, the fill operation may take some time.) To abort a fill operation, depress any key.

Figure VIII-C

ROM EMULATOR EDITOR SCREEN



D. HARDWARE CONSIDERATIONS

The ROM Emulator hardware is capable of detecting whether a ROM socket has been plugged in backwards, has not been plugged in, or has been plugged into a system with no power. The software will make checks when the user exits the ROM Emulator Editor screen. These checks, if they run into an inconsistency, will flag an error.

Selecton of ROMS to be used is made in the ROM Emulator Configuration screen. A ROM whose base address is set with "Xs" is considered unused.

If a parity error, overrun, or framing error occurs, or if an illegal character is encountered, then the transfer will immediately stop. Transfers will also be stopped by depressing any key. On download, a byte is entered into the ROM Emulator if the address of that byte is between the lower and upper address bounds entered on the ROM Emulator Transfers screen and is also being emulated, as defined by the base address of the given ROMs and the length of the device being emulated.

While editing in or transferring data to or from the ROM Emulator, it is not accessible as ROMs. Therefore, it will probably be necessary to reset the target after any of these operations.

There are two handshake methods of starting and stopping the RS-232C data stream to or from the ML4400. One method uses the RTS and CTS lines of RS-232C; the other uses DC1 and DC3 (XON and XOFF) of U. S. ASCII. The selection of these is described in Section III.F.6, "RS232C Port Settings".

The following pins are on the ML4400 rear panel Serial Port connector:

Pin No.	Signal Name (per RS-323C)	Input to ML4400	Output from ML4400
1	Safety ground	X	X
2	TXD		X
3	RXD	X	
4	RTS		X
5	CTS	X	
7	Signal ground	X	X

The ML4400 is a "DTE" device, i.e., it appears at the RS-232C interface as would a standard CRT terminal.

NOTES