Vector Lab's Microprocessor Debuggers

The Vector Lab's Microprocessor debuggers are designed as a testing aid for Atari Raster & Vector board sets in addition to other board sets that utilize similar processors. The debugger is powered from the USB port of the host computer. It communicates with the user through almost any terminal program, but a free program called TeraTerm is recommended. The communication protocol is 8 bit serial with 1 start bit and 1 stop bit at a baud rate of 38,400bps. The following describes 15 unique commands in more detail.

[1] D0-D7 BIT TEST

This test validates the proper operation of the microprocessor data bus.
The debugger prompts with the message:

```
Enter Databus Pattern >
```

The user responds with a hex value from 00 thru FF. This allows the user to probe the board under test for defective buffer IC's or shorted/open traces.
The debugger then responds with confirmation of the operation with:

```
Databus=XX
```

(It is recommended to use the Vector Labs "Test Display Board" which plugs into the 50pin card edge connector close to the microprocessor of most Atari raster and vector boards and displays address & data bus on large LED's)
After setting the data buss pattern the debugger will prompt with the message:

```
(CR)increment (E)xit >
```

The user responds with a carriage return or enter from the PC keyboard to increment the pattern by one each time. Or the user can press the E or e keys to end the test to return to the main menu.

[2] A0-A7 BIT TEST

This test validates the proper operation of the microprocessor address buss.
The debugger prompts with the message:

```
Enter Address Low Pattern >
```

The user responds with a hex value from 00 thru FF. This allows the user to probe the board under test for defective buffer IC's or shorted/open traces.
The debugger then responds with confirmation of the operation with:

```
Bits A0-A7=XX
```

(It is recommended to use the Vector Labs "Test Display Board" which plugs into the 50pin card edge connector close to the microprocessor of most Atari raster and vector boards and displays address & data bus on large LED's)
After setting the address buss pattern the debugger prompts with the message:

```
(CR)increment (E)xit >
```

The user responds with a carriage return or enter from the PC keyboard to increment the pattern by one each time. Or the user can press the E or e keys to end the test and return to the main menu.

[3] A8-A15 BIT TEST

This test validates the proper operation of the microprocessor address buss.
The debugger prompts with the message:

```
Enter Address High Pattern >
```

The user responds with a hex value from 00 thru FF. This allows the user to probe the board under test for defective buffer IC's or shorted/open traces.
After setting the Address buss pattern the debugger prompt with the message:

```
Bits A8-A15=XX
```

(It is recommended to use the Vector Labs "Test Display Board" which plugs into the 50pin card edge connector close to the microprocessor of most Atari raster and vector boards and displays address & data bus on large LED's)
After setting the Address buss pattern the debugger prompt with the message:

```
(CR)increment (E)xit >
```

The user responds with a carriage return or enter from the PC keyboard to increment the pattern by one each time. Or the user can press the E or e keys to end the test and return to the main menu.

[4] READ SINGLE ADDRESS TEST

This test reads the data from a single location of the entire 64k address space of the microprocessor.
The debugger prompts with the message:

```
Enter Address>
```

The user responds with a hex value from 0000 thru FFFF.
The debugger then responds with confirmation of the operation with:

```
Read from Address XXXX Data=XX
```

Then prompts with the following options:

```
(S)ingle (C)ontinous (E)xit >
```

Pressing the S or s key reads again from the same address. Pressing the C or c keys reads continuously until any key is hit to terminate the operation. Pressing the E or e key will end the test and return to the main menu.

[5] WRITE SINGLE ADDRESS TEST

This test writes the data to a single location of the entire 64k address space of the microprocessor.
The debugger prompts with the message:

```
Enter Address>
```

The user responds with a hex value from 0000 thru FFFF.

```
Enter Data>
```

The user responds with a hex value from 00 thru FF.
The debugger then responds with confirmation of the operation with:

```
Write to Address XXXX Data=XX
```

Then prompts with the following options:

```
(S)ingle (C)ontinous (E)xit >
```

Pressing the S or s key writes again to the same address. Pressing the C or c keys writes continuously until any key is hit to terminate the operation. Pressing the E or e key will end the test and return to the main menu.

[6] DUMP MEMORY TEST

This test dumps a block of 256 bytes of data from the microprocessor address space.
The debugger prompts with the message:

```
Enter Start Address >
```

The user responds with a hex value from 0000 thru FFFF.
The debugger responds with 16 lines of 16 bytes of data in hex along with their respective ASCII representations.
After displaying the 256 byte block the debugger then prompts with:

```
(CR)next block (E)xit >
```

The user responds with a carriage return or enter from the PC keyboard to increment to the next block of 256 bytes. Or the user can press the E or e keys to end the test and return to the main menu.

[7] SRAM TEST

This test performs a walking bit test pattern from the selected RAM memory space.
The debugger prompts with the message:

```
Enter RAM Start Address>
```

The user responds with a hex value from 0000 thru FFFF.
The debugger prompts with the message:

```
Enter RAM End Address >
```

The user responds with a hex value from 0000 thru FFFF.
*(*Note it is highly recommended that the user keep the test range as low as possible due to the nature of the walking bit test. As an example a block of 256 bytes takes approximately 60 seconds, so testing 4k of RAM will take up to 16 minutes to complete)*
When the test completes successfully then debugger will prompt with:

```
 0000 Bytes Tested (in decimal)
```

If there is an error the debugger will prompt with the message:

```
*ERROR* at address=XXXX
Data Written=XX
Data Read=XX
```

The test will terminate at the first error encountered and exit to the main menu.

[8] CHECKSUM MEMORY TEST

This test performs a checksum on the selected ROM memory space. This is particularly useful for testing program ROM's for proper data retention. Old or tarnished socketed ROM's are a major source of game failures and can quickly be verified with this test.
The debugger prompts with the message:

```
Enter ROM Start Address>
```

The user responds with a hex value from 0000 thru FFFF.
The debugger prompts with the message:

```
Enter ROM End Address>
```

The user responds with a hex value from 0000 thru FFFF.
After a small delay the debugger prompts with the message:

```
Checksum=XXXX
Bytes Summed=XXXX (in decimal)
```

The debugger prompts with the message:

```
(S)ingle (C)ontinous (E)xit >
```

Pressing the S or s key re-calculates the checksum. Pressing the C or c keys continuously re-calculates the checksum until any key is hit to terminate the operation. Pressing the E or e key will end the test and return to the main menu.

[9] FILL SRAM TEST

This test writes a user selected test pattern to the selected RAM memory block. This is a blind write to memory without any read back verification.
The debugger prompts with the message:

```
Enter RAM Start Address>
```

The user responds with a hex value from 0000 thru FFFF.
The debugger prompts with the message:

```
Enter RAM End Address >
```

The user responds with a hex value from 0000 thru FFFF.
The debugger prompts with the message:

```
Enter Fill Data>
```

The user responds with a hex value from 00 thru FF.
The debugger prompts with the message indicating test in progress:

```
Filling Memory...
```

After a small delay the debugger prompts with the message:

```
XXXX Bytes Filled (In decimal)
```

The test ends and will return to the main menu.

[10] POKEY TONE TEST

This is a pure tone test for individual channels on a POKEY chip. It is a simple audio tone test and is not a complete test of all the chips registers and capabilities.
The debugger prompts with the message:

```
Enter Pokey Base Address >
```

The user responds with a hex value from 0000 thru FFFF.
The debugger prompts with the message:

```
Enter Channel(ABCD) (E)xit >
```

The user responds with the letters A thru D to indicate which of the four channels will be tested or E to exit the test.
After the channel is selected the test routine writes a series of bytes to produce a smooth rising then falling tone to indicate proper operation. If there is no tone or choppy tones in a random fashion then that channel is probably defective. When the test is complete the debugger will go back and prompt the message:

```
Enter Channel(ABCD) (E)xit >
```

The user can then respond with the letters A thru D to indicate which of the four channels will be tested or E to exit the test.

[11] EAROM DUMP ALL

This test dumps the 64 bytes of data from the ER2055 EAROM nonvolitale memory device.
The debugger prompts with the message:

```
Enter Earom Address >
```

The user responds with a hex value from 0000 thru FFFF.
The debugger responds with 4 lines of 16 bytes of data in hex along with

their respective ASCII representations.
After displaying the 64 byte block the debugger exits to the main menu.

[12] EAROM WRITE SINGLE

This test writes a byte of data to the ER2055 EAROM nonvolitale memory device.
The debugger prompts with the message:

```
Enter Earom Address >
```

The user responds with a hex value from 0000 thru FFFF.
The debugger prompts with the message:

```
Enter Earom Data >
```

The user responds with a hex value from 00 thru FF.
If there is no error the debugger prompts with the message:

```
Data Written and verified
```

If there is an error the debugger prompts with:

```
*ERROR* at address=XXXX
Data Written=XX
Data Read=XX
```

The test will terminate and exit to the main menu.

[13] VECTOR GENERATOR TEST

This test writes a small routine to display a large “+” sign on the vector monitor via the vector RAM as a simple test of the vector generator. If the vector RAM is not stable or valid the debugger with prompt with the message:

```
**VECTOR RAM WRITE ERROR**
   TEST CANNOT PROCEED
```

As writing a corrupted routine to vector RAM and trying to execute it can damage the monitor. If there is no error message and a “+” sign is not visible or is displayed with some other weird anomalies then there are most likely other issues with the vector generator, D/A’s, op amps, or other components that drive the monitor. Again, this is a simple test and is only a validation that all is well concerning the vector generator and all its associated components.

[14] SET TO PASSIVE MODE

This command returns the debugger to the “PASSIVE MODE”. Meaning that the debugger is no longer in control of the address, data & control lines and that the target microprocessor is in complete control. None of the above described tests will be allowed to function until the debugger is returned to “TEST MODE”

[15] SET TO TEST MODE

This command puts the debugger to the “TEST MODE” mode. Meaning that the debugger is in control of the address, data & control lines and that the target microprocessor is tri-stated.

.