MEMTEST ver 1.3
By Shane Dawalt [71076,511]
For Model 4 with TRSDOS 6.x.x

Those who are familar with the Model III know that a memory tester program was provided under TRSDOS 1.3. It was a nice tool to have around when a bug crept into the system. Unfortunately, the Model 4 didn't come with a test program. The Model 4P, and possibly the 4D, have startup diagnostics which leaves the 'old' Model 4 owners in the dark. MEMTEST was written to shead some light in that dark corner of the Model 4.

MEMTEST is a program which tests all available memory in the system. MEMTEST automatically tests for the memory. MEMTEST will check and configure for:

- Main core memory (20k)
- Up to 32 memory banks (max = 32k per bank)
- Video memory (2k)
- Graphics memory (32k)

Program Execution

MEMTEST can be forced to output data to the printer as well as the video. Simply entering "MEMTEST<enter>" on the command line will cause output to flow to the video. Entering "MEMTEST P<enter>" will cause output to flow to the video and printer.

User Interaction

The user need only read the screen as the testing continues. <BREAK> may be pressed at any time to escape from a memory test. Upon exiting by a <BREAK>, or completion of testing, the system will be replaced to its original condition prior to the execution of MEMTEST thus preventing the familar reboot which was a tradition with the Model III tester.

Screen Output

At initialization, the screen will be cleared and banner information will be printed. During testing, the memory area name will be displayed and a series of numbers will be displayed directly below the memory area name. The numbers are the most significant byte of the address being tested. These numbers are displayed in 2K intervals regardless of the memory area being checked. Note that an address relative to 0 is generated for the testing of the graphics board. This is generated by multiplying the y coordinate by 128 then adding the BYTE value of the x coordinate.

If a byte error occurs during testing, an error line is displayed at the bottom of the screen along with the total number of errors for that particular memory area. (After testing of each memory area is completed, the

error counter is zeroed.) The error line contains the address where the error occured, the test byte used and a bit map of all errored bits returned. The bit map is a series of numbers from 7 to 0 where 7 is the most significant bit and 0 is the least siginficant bit. Only the numbers which are displayed are the actual errored bits. The errored bits are the bits which are logical 1 after testing. Those which are not displayed are correct. To determine what the original byte returned from the memory was, Exclusive OR (XOR) the bit map with the test byte.

A typical error line output to the screen would look like this:

```
Error at address 7F08x.  Test byte: 55H  Errored bits:    6 5     2 1 0
Error count =      2
```

This means that the byte used to test location 7F08x was 55H. That returned from the memory location at 7F08x must have been 01010101 XOR 01100111 = 00110010 or, for those of you who like hexidecimal, 55h XOR 67h = 32h. This means that bits 6,5,2,1 and 0 are errored (a rather nasty memory problem!). The error count simply means that a previous memory error has occured. The current total of memory errors is 2.

Hardcopy Output

A hardcopy of errors can be obtained if a printer is connected to the system. Refer to "Program Execution" on how to declare a hardcopy output. The ouput will be essentially the same as the screen. The only difference is when testing is completed for a particular memory area and if no errors occured, "No errors occured" will be printed on the page. Also, the "Error count" message will be displayed at the end of the normal message line on the printer.

Other things

MEMTEST does not destroy memory contents nor does MEMTEST destroy the bank usage flags of TRSDOS. Because of this, applications may use, or be linked to, upper banks during testing without fear of a system crash or lost data and the conclusion of testing. Memory will be destroyed between 2600h and 3000h. This is reserved for program loading/usage.

*** WARNING *** WARNING *** WARNING *** WARNING *** WARNING ***

MEMTEST shuts down the RTC interrupt during the testing process. The real time clock will not be accurate after MEMTEST has completed testing. Since the RTC interrupt is extinguished, the cursor will not flash. If a printer error occurs, the computer will seem to freeze up. MEMTEST is waiting until DOS returns control. During this time, no keyboard input is possible, i.e. the <BREAK> key will not exit the program. After about 10 seconds, MEMTEST will resume normal operations.

*** WARNING *** WARNING *** WARNING *** WARNING *** WARNING ***

Finally

MEMTEST was developed and tested on a Model 4 with 128K RAM and an RS graphics board installed. MEMTEST tested the entire 153.6K of RAM space in just over 30 seconds (without hardcopy output). This timing check will not hold if the harcopy option is selected. For the skeptics, MEMTEST checks each memory byte with 4 different masks before continuing.

A SPECIAL NOTE:

> When testing the graphics memory, be aware that a fault in the memory DOES NOT necessarily imply a bad RAM location. The Tandy graphics board is build using the idea that 4, 64K RAMs can be used like 8, 32K RAMs. (This is somewhat comparable to the bank memory where 8, 64K RAMs are made to look like 16, 32K RAMs). A gate array and various PAL chips are implemented to convert the 4 bit data from the 4, 64K RAMs to 8 bit data and visa-versa. Along with doing this bazaar conversion, the gate array is in charge of the storing of the options register data, storing of both x and y coordinates and taking care of the data register. As can be seen, a flaw in the gate array could render any I/O port to the graphics board and/or any RAM location faulty, whether it is or not. MEMTEST does not try to interrogate the graphics board to decide where the error is if one occurs (this is a job for the tech.).

Comments or questions may be sent to my EMAIL box or messages may be left on the LDOS forum. For forum messages, please mark them with my ID number listed in the heading of this document.

Shane Dawalt