

Mini Memory Tester

Alpha Documentation

By Dagen Brock

Table of Contents

[Table of Contents](#)

[Main Menu](#)

[Testing](#)

[Test Address Ranges](#)

[Running Tests](#)

[Memory Errors](#)

[Generating memory corruption](#)

[Missing Features](#)

Description

Mini Memory Tester is a simple testing program for Apple IIgs memory cards.

Main Menu

Here's a look at the main screen with the menu options on the left:



Using the up and down arrows, you can select which menu item you want (indicated with square brackets), and hit enter to edit it, or if "BEGIN TEST" is selected, it will of course begin testing memory using your specified parameters.

When entering values, it expects hexadecimal numbers (0 through F). You do not need to hit enter. Text entry is complete when you type the appropriate length of hexadecimal number. You can hit ESC to cancel the current number entry.

Parameters should be fairly obvious:

- Start Bank - End Bank
 - Range of memory banks that will be tested
- Start Addr - End Addr
 - Hex address to test at within banks
- Test Byte
 - Hex value that will be used to write during test
- Test Delay
 - Delay in seconds (HEX!) between passes

Testing

When you begin test, the program will write your specified test value to all memory locations specified, wait the desired period, and then read the same locations looking for your value. If the wrong value is read, it will generate an error (see Errors).

Test Address Ranges

If you specify a range of, say “2000-8000” across multiple banks, then it will only test that range. Example, if you say banks 10 through 11, with the address range above then it will only test:

10/2000 ... 10/8000

11/2000 ... 11/8000

Just be aware that it's not contiguous between banks.

Running Tests

Running tests show the current pass, and the current address chunk being written or read, and a timer if it is waiting for the next pass.



After each pass of writing and reading, the pass number is incremented, and the next pass starts.

Note: Memory Data is not scrambled between passes!

Note: If you press a key, the test will cancel at the end of the current Write/Read pass.

Memory Errors

When the test encounters an error, it will show an error message. You can press a key to continue testing. It will leave the memory error during testing until another error is encountered or the test is cancelled.

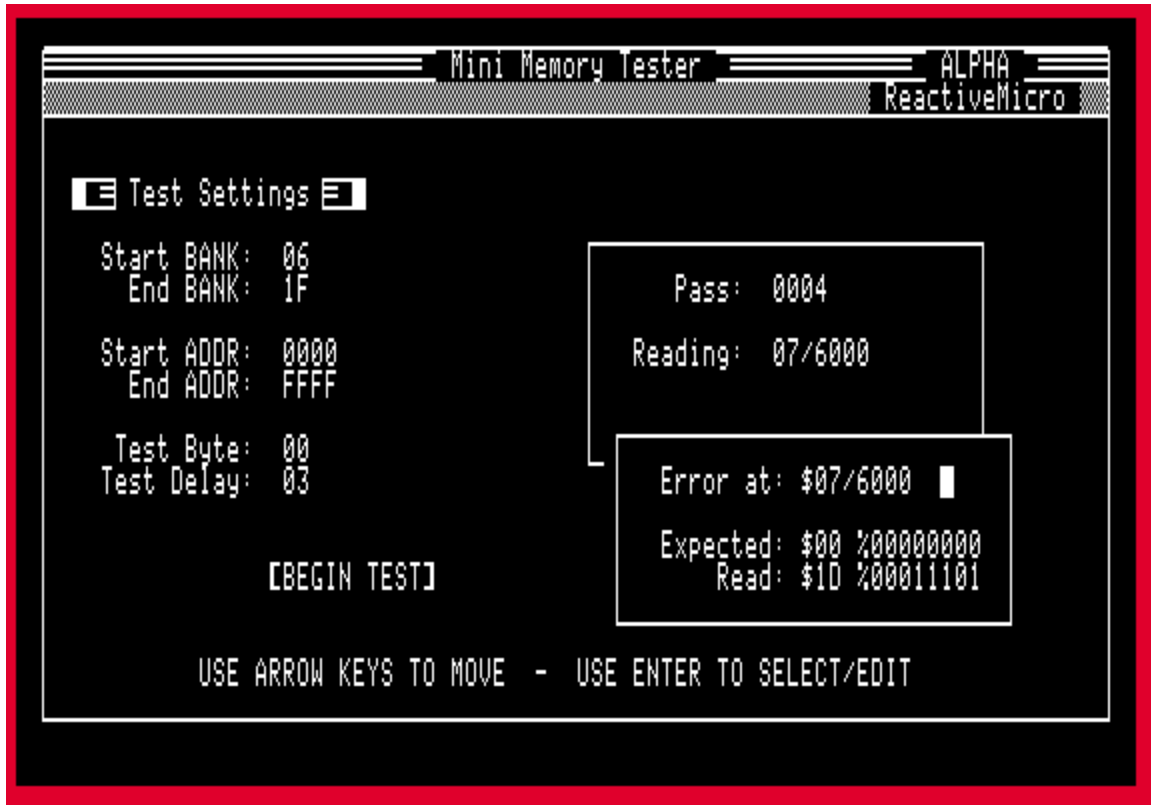
Generating memory corruption

It is helpful (particularly to test the program itself!) to be able to corrupt some memory during the test to see that things are working properly. To do this, simply press “c” (lowercase) to “corrupt” the current spot being written. This will change the border color to let you know that some memory was changed and should generate an error when the Read pass occurs.

You can see the border color has changed to red after I hit “c” during this test.



Then during the read phase, it caused this error:



As you can see, it shows the error location, the expected value (in hex and binary) and the value we got when we read that location.

Missing Features

As an alpha release, I made a conscious decision to “get something out the door” versus making people wait for perfection (which will surely never come).

Some things that I think are missing:

- Allow user to set the writer to use pseudo-random values as well as just a simple Test Byte.
- Full 16-bit version of test. I think this might be important to find things that might not be visible on only the 8-bit data lines.
- Wiping behaviors. Allow user to specify some sort of wipe between passes.
- Count total errors. Might be nice.
- Checkerboard writes. Maybe write 00,FF,00,FF,... on one pass, then FF,00,FF,00,... on the next pass. Possibly less important if Wiping behaviors are supported.