Multi RAM Tester

Version 1.0

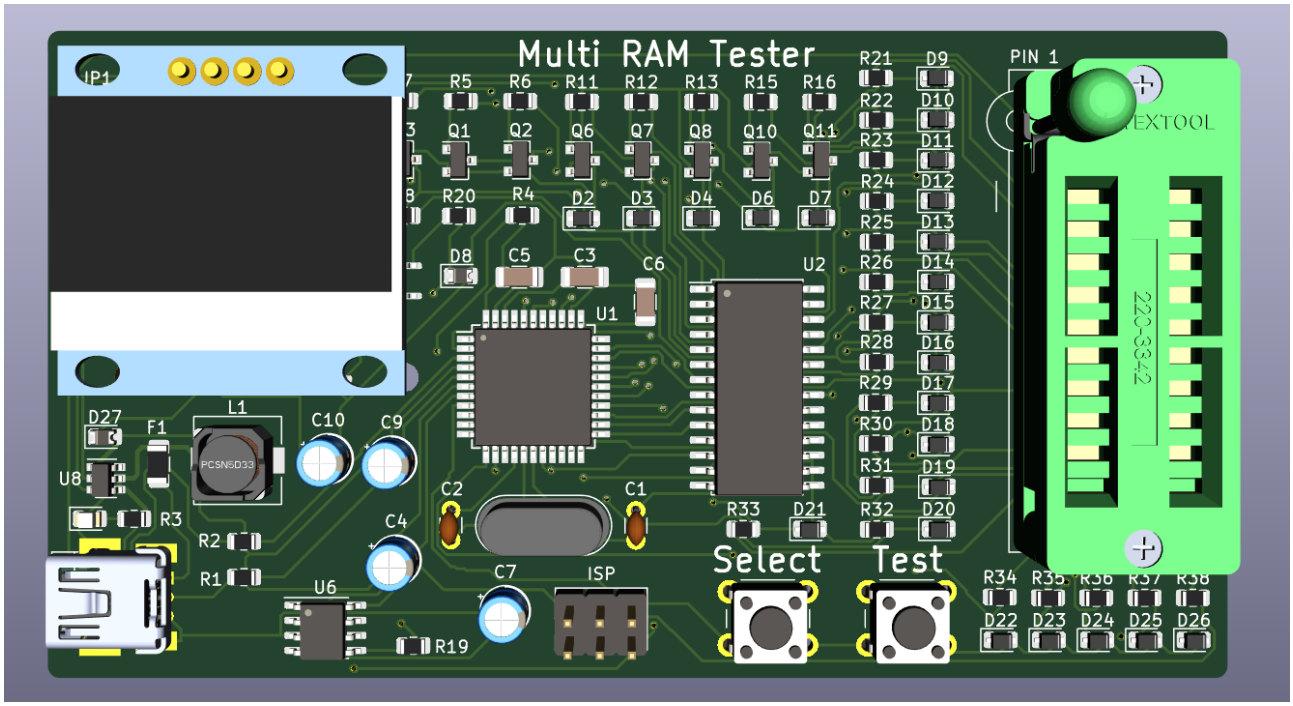


Table of Contents

1. Introduction

The Multi RAM tester was developed in order to test vintage RAM chips from the 70's and 80's. Modern testers like the TL866 do not check these chips due to them being long obsolete.

The idea for this project started when I needed to test a few RAM chips for the ZX Spectrum and had to hack together a Arduino based tester. There are lots of schematics and programs available on GitHub to allow a DIY'er to do this.

After needing to build another Arduino based tester to test different RAM chips for the Commodore 64, I decided to design a relatively inexpensive Multi RAM tester to check most of the common RAM chips that the TL866 does not support (The Multi RAM Tester does therefore not test the larger chips like the 28 pin 62256 or 6264 as this would require Atmel Mega2560 which would increase the cost, and the TL866 also tests these chips)

This device includes a -5V converter and 12V boost circuit to allow for testing on the 4116 chips which require +5V, -5V and 12V.

While this device checks all the common RAM chips, if there are other types of RAM that users need adding (and the hardware supports it) I will update the firmware and make available for download. The device can then be upgraded via the USB port using avrdude (see Section 5).

Please read though this manual completely before testing any chips so that you know the correct operating procedure in order to avoid damaging any chips under test.

2. Changelog

- V1.0 - First release 1 July 2021

3. Supported chips

The tester supports the following DRAM and SRAM chips (Up to maximum of 20 pins, narrow)

SRAM Chips		
AM2700, AM2701	AMD	256 x 1
D3106, D3107	Intel	256 x 1
4256	Motorola	256 x 1
MM74C200	National	256 x 1
N8216, **8217**, 82116, 82117	Signetics	256 x 1
SN74200, 74201, 74300, 74301	Texas Instruments	256 x 1
AM9102, AM2102	AMD	1k x 1
D2102, D8102	Intel	1k x 1
NTE2102	NTE	1k x 1
TMS4033, 4034, 4035	Texas Instruments	1k x 1
AM2148, 9114, 9124	AMD	1k x 4
CY2148, 2149, 7C148, 7C149	Cypress	1k x 4
MB8114, MB8148, MB8149	Fujitsu	1k x 4
HM4334, HM472114, HM6148	Hitachi	1k x 4
D2114, D2148, D2149	Intel	1k x 4
TMM314, **TMM2114**	Toshiba	1k x 4
TMS2114, 4014, 4045	Texas Instruments	1k x 4
CY7C167	Cypress	16k x 1
MB8167	Fujitsu	16k x 1
HB6267	Hitachi	16k x 1
IDT6167	IDT	16k x 1
MCM2167	Motorola	16k x 1

DRAM Chips		
AM9016	AMD	16k x 1 (-5V, 5V, 12V)
MB8116, 8126, 8216	Fujitsu	16k x 1 (-5V, 5V, 12V)
2116, 2117	Intel	16k x 1 (-5V, 5V, 12V)
IM4116	Intersil	16k x 1 (-5V, 5V, 12V)
MCM4116	Motorola	16k x 1 (-5V, 5V, 12V)
uPD2116, uPD416	NEC	16k x 1 (-5V, 5V, 12V)
NTE2117	NTE	16k x 1 (-5V, 5V, 12V)
TMS4116	Texas Instruments	16k x 1 (-5V, 5V, 12V)
MB8118	Fujitsu	16k x 1
HM4816	Hitachi	16k x 1
2118	Intel	16k x 1
MCM4516, MCM4517	Motorola	16k x 1
MB81416	Fujitsu	16k x 4
TMS4416	Texas Instruments	16k x 4
MB8264	Fujitsu	64k x 1
HM4864	Hitachi	64k x 1

2164A	Intel	64k x 1
MCM4164, MCM6664, 6665	Motorola	64k x 1
uPD4164	NEC	64k x 1
TMS4164	Texas Instruments	64k x 1
TMM4164	Toshiba	64k x 1
MB81464	Fujitsu	64k x 4
HM50464	Hitachi	64k x 4
MT4067	Micron	64k x 4
M5M4464	Mitsubishi	64k x 4
KM41464	Samsung	64k x 1
TMS4464	Texas Instruments	64k x 4
MB81256	Fujitsu	256k x 1
HM50256, HM51256	Hitachi	256k x 1
21256	Intel	256k x 1
MCM6256	Motorola	256k x 1
uPD41256	NEC	256k x 1
TMS4256	Texas Instuments	256k x 1
TMM41256	Toshiba	256k x 1
MCM514256	Motorola	256k x 4
KM44C256	Samsung	256k x 4

(The chips in **bold** have been tested on the tester. As the other chips are all equivalents they should also work but have not been tested. Please feedback to me if you have tested these so I can update this document)

When testing compatible chips always check the datasheet to make sure the chip is pin for pin compatible, especially the power pins. The chip under test could be damaged if the incorrect type is selected when testing.

4. Test Procedures & Limitations

The test procedures are designed to run quickly and look for common issues with RAM chips.

While the tester does a good job at detecting faulty memory cells and stuck data/address lines, it will not detect every single possible issue 100% of the time. A product like this is unlikely to exist especially when testing 20 year old, or ever 30 year old memory chips.

As the tester uses a Atmel 32u4 it cannot test the RAM at their full rated speed (RAM Speed could be down to 25ns.) To test chips at their fastest speed a much faster microcontroller would be needed which would make this tester much more expensive and not really affordable for the hobbyist.

The Multi RAM Tester also does not measure the power used by the chips under test, so you could for instance have a ‘faulty’ chip that draws a low of power but still functions correctly when reading/writing memory.

The Multi RAM Tester performs 3 tests:

- Test for stuck address lines (high or low)
- Test for stuck data lines (high or low)
- Test each memory location using a variety of test methods (walking, fill, random)

5. Upgrading

The Multi RAM Tester is software upgradable however this requires some knowledge on using avrdude, and Arduino IDE (to re-flash the bootloader if needed)

You could brick the device if you attempt to upgrade with the incorrect settings. (Though it is possible to recover by reprogramming the Atmel32u4 via ISP)

To reprogram the tester we will make use of avrdude which is included in *Arduino IDE*.

5.1 Upgrading Firmware

Tools required:

- Latest Multi RAM Firmware. https://myretrostore.co.uk/mrt
- Arduino IDE https://www.arduino.cc/en/software (Linux, Windows and Mac versions are available)

Install Arduino IDE (If using Windows, install all the device drivers during the install)

Using a mini USB cable, connect the Multi RAM Tester to the USB port.

To reprogram the ATMega32u4 with avrdude, you first need to send a reset command via the serial port which then restarts the microcontroller and is then ready for programming.

Windows: See https://github.com/p1ne/arduino-leonardo-uploader

Linux: See https://github.com/nicholaskell/Arduino_Loader

Once the microcontroller has been reset, you can then upgrade the flash:

avrdude -p atmega32u4 -C /etc/avrdude.conf -c avr109 -b 57600 -D -P "/dev/ttyACM0" -U flash:w:firmware.hex:i

On Linux I see the following script to upgrade the flash:

https://myretrostore.co.uk/mrt-upgrade.txt

The output should be similar to the below:

Connecting to programmer: .

Found programmer: Id = "CATERIN"; type = S

Software Version = 1.0; No Hardware Version given.

Programmer supports auto addr increment.

Programmer supports buffered memory access with buffersize=128 bytes.

Programmer supports the following devices:

Device code: 0x44

avrdude: AVR device initialized and ready to accept instructions

Reading | ## | 100% 0.00s

avrdude: Device signature = 0x1e9587 (probably m32u4)

avrdude: reading input file "firmware.hex"

avrdude: writing flash (24818 bytes):

Writing | ## | 100% 2.13s

avrdude: 24818 bytes of flash written

avrdude: verifying flash memory against firmware.hex:

avrdude: load data flash data from input file firmware.hex:

avrdude: input file firmware.hex contains 24818 bytes

avrdude: reading on-chip flash data:

Reading | ## | 100% 0.39s

avrdude: verifying ...

avrdude: 24818 bytes of flash verified

avrdude: safemode: Fuses OK (E:CB, H:D8, L:FF)

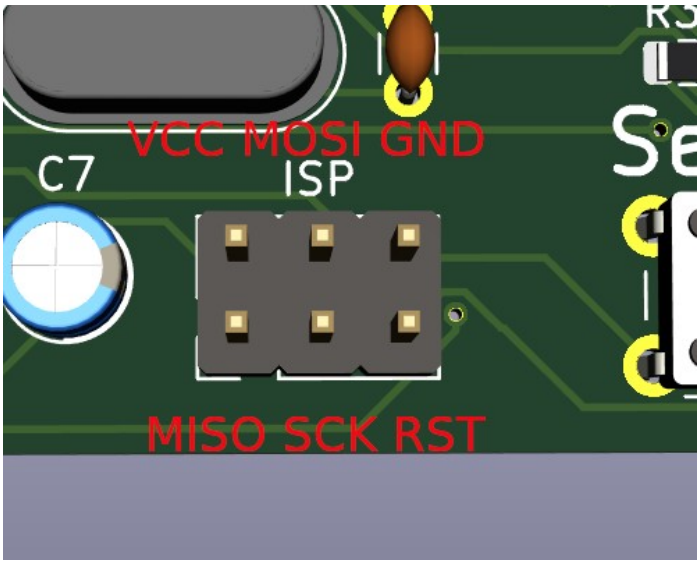
avrdude done. Thank you.

5.2 Re-programming bootloader

There should be reason re-flash the bootloader, but if you have by mistake managed to corrupt the flash, the bootloader can be reprogrammed. To re-flash the bootloader you will need to use an ICSP programmer like the USBTinyISP, USBasp, 'Arduino as ISP', or other hardware that the Arduino IDE supports.

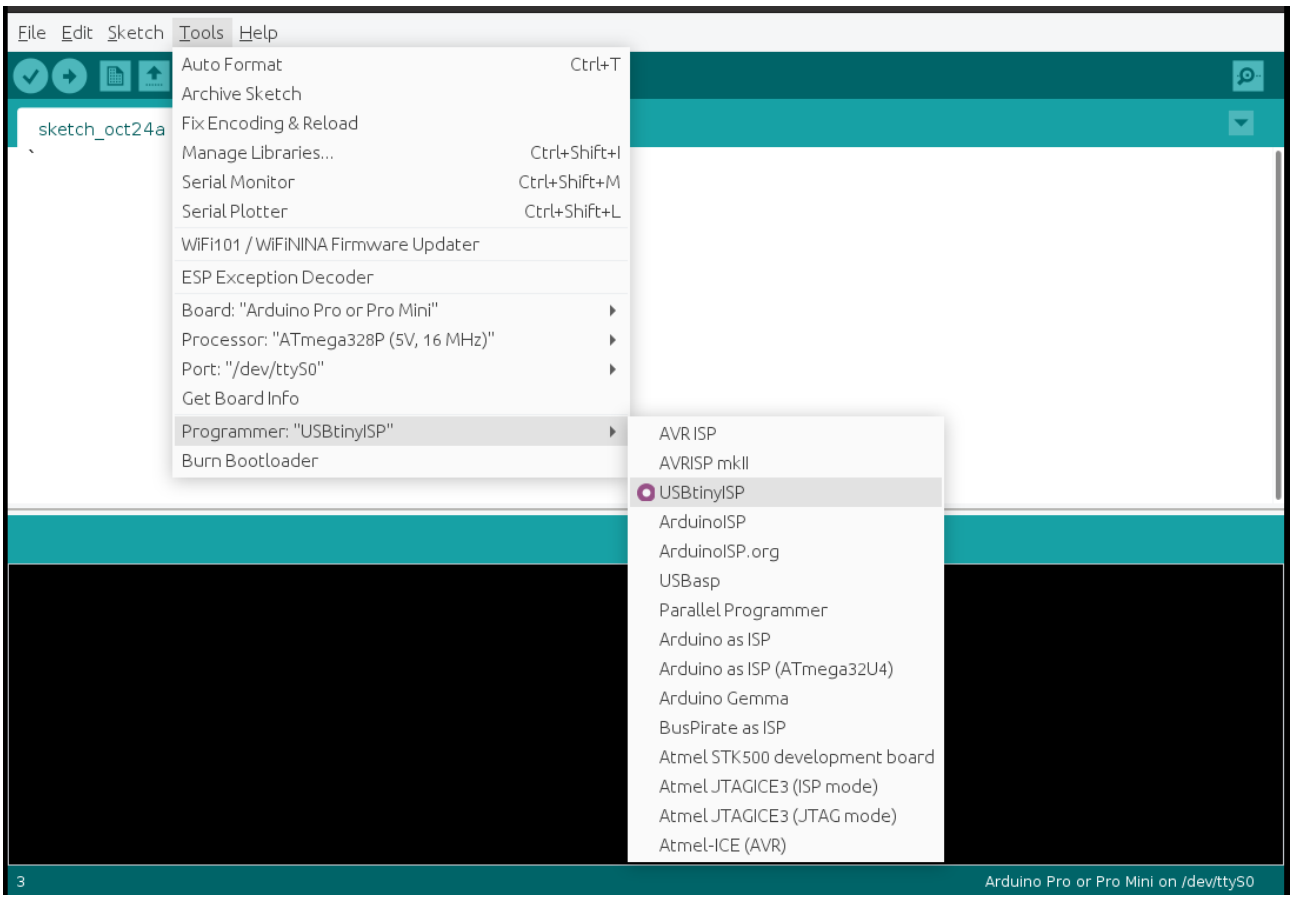
Connect the ISP header from your programmer to the ISP pins on the Multi RAM Tester.

Take note of the pinouts. Connecting the pins up incorrectly and you risk damaging the tester and the ISP programmer.



Run Arduino IDE

Under **Tools / Programmer** select your programmer.



Select the programmer port: **Tools / Port**

Select the correct board: **Tools / Board / Arduino Leonardo**

Then program the bootloader: **Tools / Burn Bootloader**

6. Usage

To power the device you will need to use a USB Mini Type-B cable and power if from a suitable USB device that can supply 5V with at least 1A current.

After power up the display screen will be initialised and the version number will be displayed for 2 seconds.

Press the *Select* button to scroll through the list of chips to test (See Section 3 for a list of compatible chips)

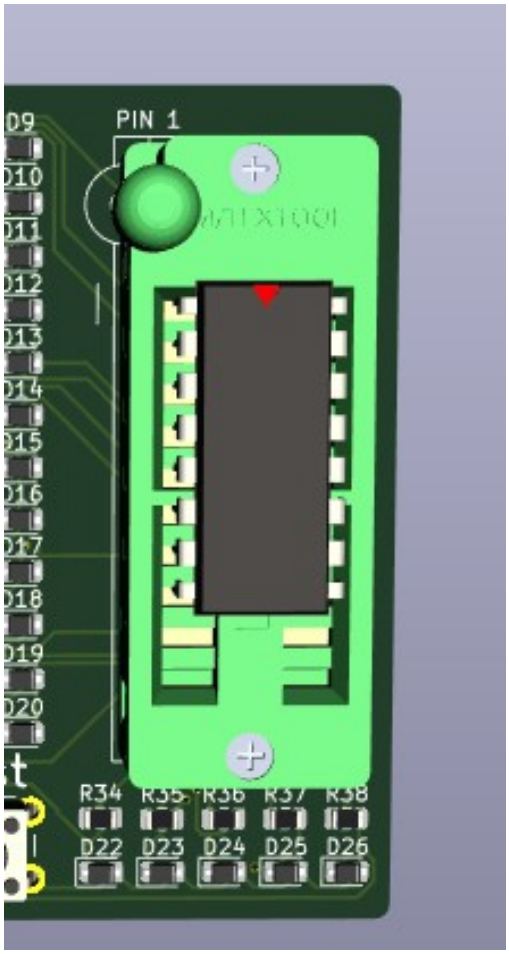
RAM	Test time
8217	15s
2102	19s
2114	40s
6167	55s
4116	10s
4816	12s
4416	30s
4164	35s
41464	2 mins
41256	2 mins
44256	7 mins

When testing compatible chips always check the datasheet to make sure the chip is pin for pin compatible, especially the power pins. The chip under test may be damaged if the incorrect type is selected when testing.

Press *Test* button to select the chip to test. The tester will then prompt you to insert the chip in the ZIF socket.

Take note of the chip orientation. Failure to do so and you risk damaging the chip.

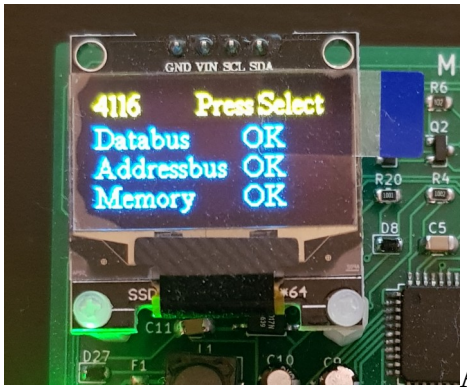
For all RAM chips under test, pin 1 is always on the top left.



Press *Test* button to begin the test

The device will then perform tests on the data pins, address pins, and memory.

- Databus – tests for stuck HIGH or LOW pins on the data bus
- Addressbus – tests for stuck HIGH or LOW pins in the address bus
- Memory – tests all memory locations using fill, walking and random memory locations.



A failure in any of these tests indicates that the chip is faulty and needs to be replaced.

After the test has completed the voltage is remove from the power pins, and can then be safely removed from the ZIF socket.

Press *Select* button to go back to the chip select menu (The last chip type tested is saved to the devices EPROM so that when you next apply power to the device it will default to the last chip tested)

6. Diagnostic Mode

The Diagnostic mode was created to allow testing of the device when building to ensure all the ZIF pins are functioning as expected. This is a useful check should your tester suffer any hardware damage which can then be diagnosed.

Tools required:

- Digital Multimeter

To enable the Diagnostics Mode make sure power is removed from device.

Hold down the “Test” button and power up the device. After 2 seconds, the device will enter diagnostics mode.

The following is then performed in a continuous loop. You will need to use a DMM to measure the voltages on the ZIF pins. There is a 3 second delay between each state change to allow for time to measure the voltages.

To measure the I/O lines, -5V, +5V, +12V you can connect your ground from the DMM to the shield of the USB connector (This is GND)

To measure when the pins get set to GND, connect your ground from the DMM to the GND pin displayed on the OLED, and the red lead to VCC pin as displayed on the OLED

- All I/O lines are pulled high, then low. (Pins 1-9, and 11-20. Pin 10 is only used for 5V/GND)
- -5V enabled on pin 1
- 12V enabled on pin 8
- 5V enabled on pins 8, 9, 10, 13, 14, 20
- GND enabled on pins 8, 9, 10, 13, 20

To exit diagnostics mode, simply remove power from the device for a few seconds and power up again.

7. Contact

If you have any technical queries, feedback, bug reports or wish lists, you can contact me at sales@myretrostore.co.uk