

Informations about the „SYSTest64+OLED“ Boards

(Final Prototype)

Last Changes

Donnerstag, 19. Januar 2023

Table of Contents

1 Advantages of the SYSTest64 + OLED DIAG Cartridge.....	2
2 The Operation.....	3
2.1 OLED Display.....	4
2.2 ROM Type Normal/Ultimax.....	5
2.3 Firmware Switch.....	5
3 The Tests.....	6
3.1 ALL.....	6
3.2 MAIN CHIPS.....	6
3.3 LOW MEMORY.....	6
3.4 SYSTEM TEST.....	7
4 The additional programs in ROM (W27C512).....	8
4.1 PERIPHERALS.....	8
4.2 SID ADDRESS.....	9
4.3 DISPLAY.....	10
4.4 MIDI.....	11
5 The Harness Adapter.....	12
5.1 Using SYSTest64 and DIAG 586220 software simultaneously.....	14
6 Dongles.....	15
7 The LED boards.....	16
8 „MAIN CHIPS“ Test Evaluation.....	17
9 „SYSTEM“ Test Evaluation.....	18
10 Change the names of ROM 7 and ROM 8.....	19
11 Pro Mirco .hex File (Firmware) transmission.....	20
11.1 Firmware problem - flash bootloader (optional).....	22

1 Advantages of the SYSTest64 + OLED DIAG Cartridge

Until now you had to use a lot of different hardware to get to the bottom of possible hardware errors of a C64. And some things were completely impossible before the SYSTest64 + OLED cartridge. Especially when the C64 does not start and does not display a picture. Then usually an oscilloscope is needed. Already here the new cartridge steps into the breach, checks the most necessary signals, and displays the result on the OLED display.

Furthermore, the cartridge offers a menu that can be operated with only one button, and thus various hardware tests can be conveniently selected and started.

In addition, there is a whole series of additional programs which are stored on the 64K large 27C512 EEPROM and are started as independent **ROMs** via the menu.

Here is an overview of the position of the ROMs on the 64K W27C512 EEPROM:

- 1: SYSTEM Test
- 2: LOW MEMORY Test
- 3: PERIPHERALS Tests
- 4: SID ADDRESS Test
- 5: DISPLAY Test
- 6: MIDI Test
- 7: FREE (for your own ROM)
- 8: FREE (for your own ROM)

2 The Operation



The **menu** is operated with only one **button**.

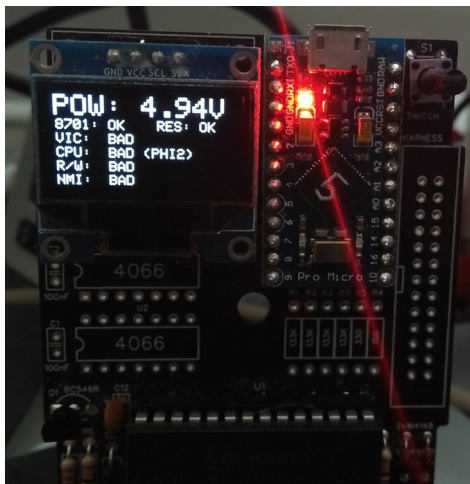
A **short press** navigates down in the menu. Once the lowest menu item is reached, the navigation star jumps all the way up again.

A **long press** is used to switch to rotation mode (3 arrows to the right then appear on the left). In rotation mode, you can rotate through the options by briefly pressing the button. If the button is pressed again for a long time, the rotation mode is switched off again.

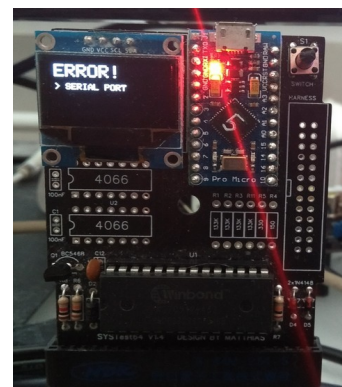
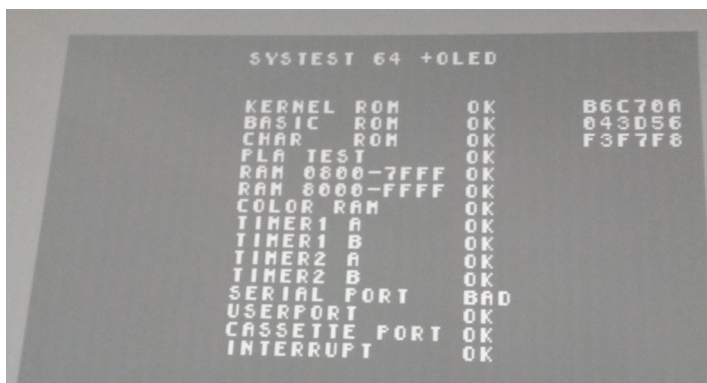
A **long press** on the button while the selection is on the „Start“ or „Run“ menu items, starts the selected Test („Start“) or ROM („Run“).

2.1 OLED Display

The OLED 1306 display (128x64 Pixel, 0.96 inch) is one of the important things about SYSTest64, as it allows a good overview of the C64's functionality even when the C64 has a "black screen" (no picture to be seen).



The OLED display also shows the same error (the first one) in the main test:



So you don't necessarily need a connected Monitor/TV to determine if the C64 is working properly.

2.2 ROM Type Normal/Ultimax

For normal ROM modules the ROM Type "Normal" must be selected in the menu. If you use for example the Commodore DEAD Test ROM, the ROM Type "Ultimax" must be set, because this module was programmed for the Ultimax Mode.

2.3 Firmware Switch

If the button is held down during power-on for 1 second, the firmware can be changed. Which firmware is active is briefly displayed at the bottom right.

3 The Tests

3.1 ALL

In this test, all existing tests are performed

3.2 MAIN CHIPS

The voltage of the system is tested, as well as essential signals which are necessary for the C64 operation.



3.3 LOW MEMORY

The memory range from 0-7FF is checked. For this purpose, the ULTIMAX mode is switched. This mode does not require KERNEL, BASIC, CHAR ROMs. So it can be tested well if the system works without the before mentioned ROMs. If this is the case and the following system test does not start, the ROM blocks should first be checked/replaced.

3.4 SYSTEM TEST

This is the most comprehensive test, whereby all possible tests are performed here



Starting from ROM tests, up to external port tests.

To test the keyboard and joystick ports, the ROM: "PERIPHERALS" can be selected and started in the menu (see below).

4 The additional programs in ROM (W27C512)

4.1 PERIPHERALS



With this ROM program the keyboard and the joystick ports can be tested together on one screen for the first time. Also the test of a mouse in **Joystick Port 1** is possible.

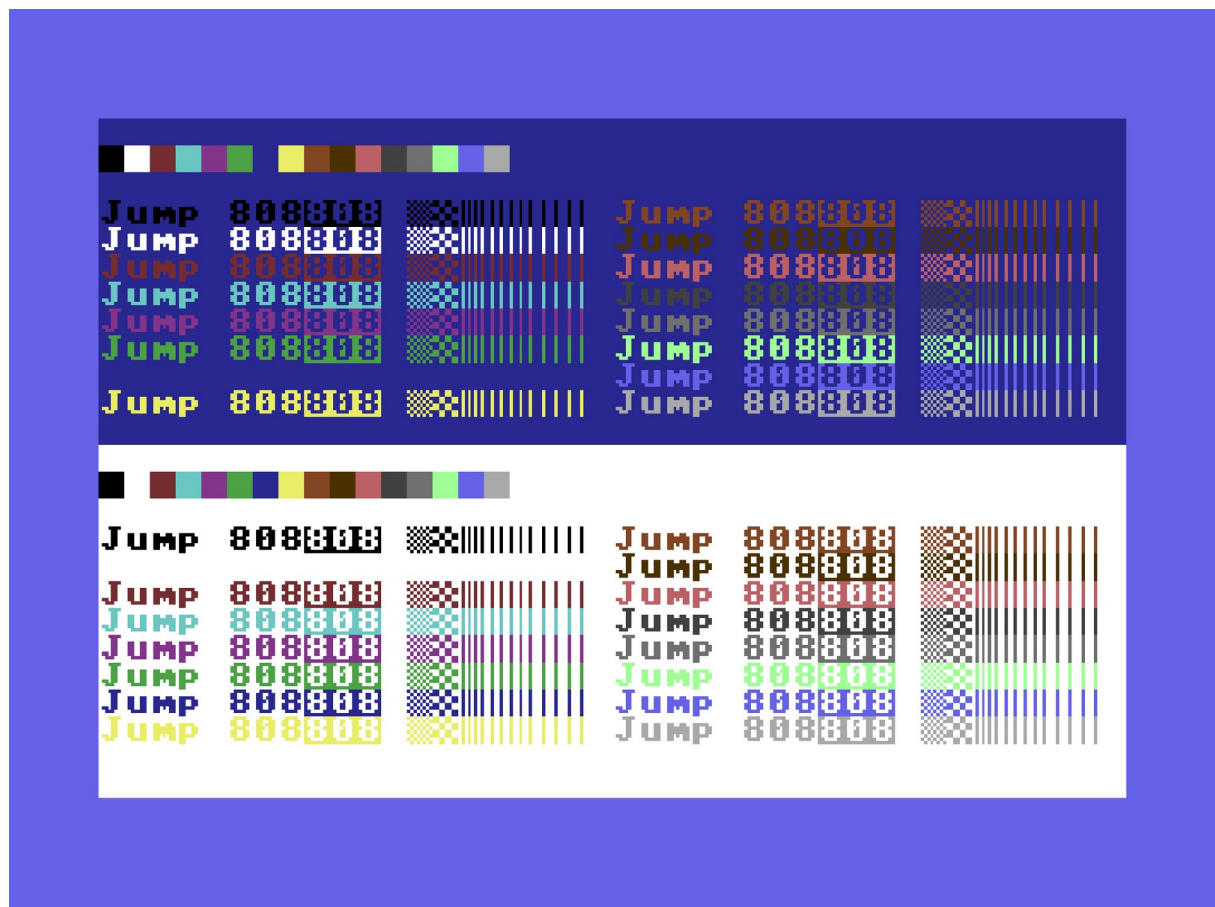
4.2 SID ADDRESS



This ROM program displays all SIDs / SID addresses available in the system (white). The cursor keys can be used to navigate through the addresses. The currently selected address is highlighted by the yellow text color.

Pressing SPACE or RETURN will sound a tone from the selected SID.

4.3 DISPLAY



This ROM program can be used to evaluate the video signal quality. Depending on how well the video signal is displayed on the monitor, you can see sharp or blurred contours.

4.4 MIDI



This ROM program can be used to test a connected MIDI interface and MIDI keyboard.

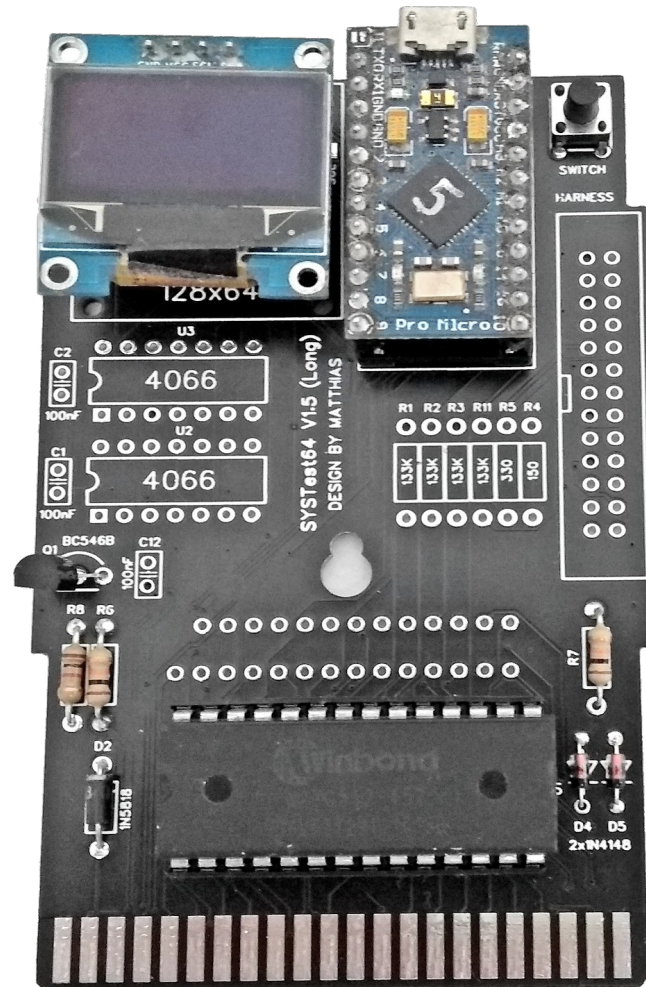
5 The Harness Adapter

The harness is compatible to the common harness adapters for the C64.



However, this is not absolutely required for the SYSTest64+OLED board and is only supported for compatibility reasons (for the old DIAG Software by Commodore). If the old DIAG Software by Commodore and the Harness Adapter is not used, only the three dongles **USERPORT**, **TAPE-PORT** and **SERIAL** are required. When using the Harness, the dongles for the **USERPORT** and **SERIAL**, as well as a **TAPE-PORT adapter** with post socket are additionally required.

Without the harness, and with the dongles, most of the components on the board are not needed:



With this, SYSTest64 works perfectly.

5.1 Using SYSTest64 and DIAG 586220 software simultaneously

If you do not want to create two TAPE port dongles for using the old DIAG 586220 software **with harness** and for using the new SYSTest64 software **without harness**, there is a simple solution for this:

1. On the SYSTest64 board do not solder the 330 Ohm (R5) and 150 Ohm (R4).
2. Solder the 330 Ohm and 150 Ohm resistor and bridge on the TAPE dongle.

6 Dongles

USERPORT

B → 8

C → H

4 → 6

D → J

5 → 7

E → K

F → L

9 → M

TAPE PORT

6 → 4

5 → 150 Ohm → 3

5 → 330 Ohm → 1

SERIAL PORT

1 → 5

3 → 4

KEYBOARD (is not required with the SYSTest64 board)

5 → 17

6 → 14

7 → 15

8 → 16

9 → 20

10 → 18

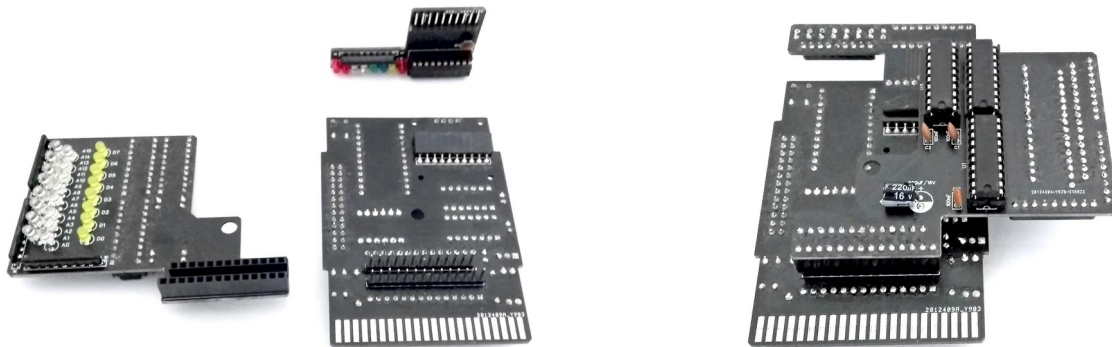
11 → 19

12 → 13

7 The LED boards

The LED boards are not required for the tests. However, these can already provide information about whether the system is working correctly. Because with them the signal flow of the address and data bus can be observed.

Here you can see how the LED boards can be plugged and attached to the back of the SYSTest64 board. The upper LED board is inserted from above into the SYSTest64 board socket. The left LED board is plugged straight onto the SYSTest64 board.



Note on equipping the LEDs and 74HCT573:

The LEDs should be assembled last. Each LED has a long and a short wire. The question is how to solder the LEDs. This can be easily found out as follows. Put the assembled boards together, and then plug the finished SYSTest64 board into the C64 expansion port via a 90° angle adapter, and power up the C64. Now an LED is plugged into one of the LED double holes. If the LED lights up, all LEDs in this row can be soldered in this way.

Instead of the **74HCT573**, **74HC573** can also be used.

8 „MAIN CHIPS“ Test Evaluation



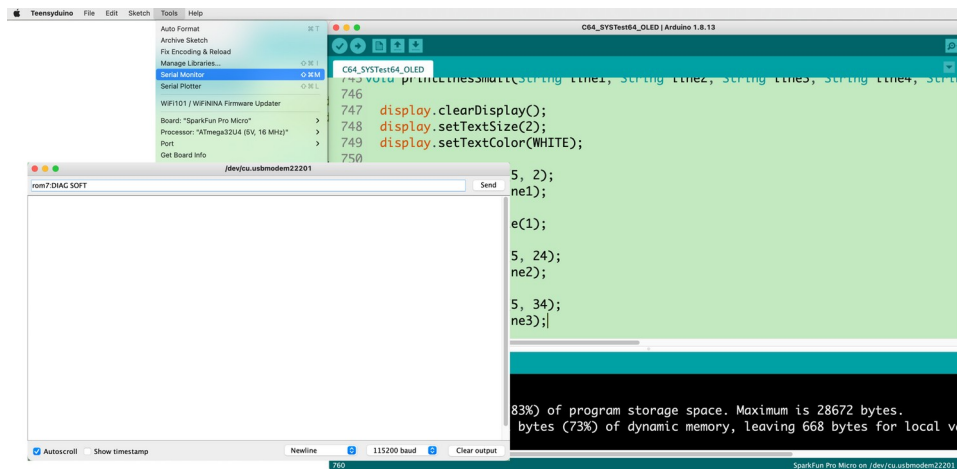
If a "BAD" should appear on one of the chips, then the priority is from top to bottom.

If, for example, the 8701 chip is defective, all other chips further down will also display "BAD". Therefore, troubleshooting should **always start with the "topmost BAD"**.

9 „SYSTEM“ Test Evaluation

Important to know is, if "Userport" or "Cassette Port" show Bad, then "Interrupt" is automatically Bad too.

10 Change the names of ROM 7 and ROM 8



The names of ROM 7 and ROM 8 can be changed and permanently stored via the USB interface (Serial Port).

Every program that opens a serial connection via USB (Serial Port) with 115200 Baud can be used.

You can also open the Serial Monitor of the Arduino IDE (the connection data is shown in the status line of the Serial Monitor). Then only one line has to be sent. For example:

rom7:DIAG Software

or

rom8:DEAD TEST

(the Pro Micro must respond to the commands with the same line!)

Afterwards, the USB plug must be removed and the SYSTest64 must be switched off. Then the new settings will be used at the next power up.

11 Pro Mirco .hex File (Firmware) transmission

There are several ways to transfer the .hex file to the Pro Micro. However, a USB cable is always required. This is used to connect the Pro Micro to a PC.

Transfer via software with GUI:

The easiest way to transfer the .hex file to the Pro Micro is undoubtedly a program with a graphical user interface, in which the .hex file is selected, and then transferred by pressing a button.

Transfer via software without GUI:

Another way is via the tool "avrdude" which is operated via the command terminal. For this, avrdude must be installed on the PC beforehand.

Here is an example of **how it works on a Mac**

If "avrdude" is installed and a terminal window is open, entries can be made in this terminal window.

First we search for the correct USB port to which the Pro Micro is connected. For this we enter the command "ls dev/cu". Then a list is displayed:

```
/dev/cu.Bluetooth-Incoming-Port
```

```
/dev/cu.usbmodem241401
```

```
/dev/cu.usbmodem241201
```

To find out which USB port (in this case a device = "dev") is the correct one, the Pro Micro should be removed from the USB port. Then enter the command "ls dev/cu" again and execute it. The USB port that is now missing is the Pro Micro USB port. Then reconnect the Pro Micro.

As an example we take `"/dev/cu.usbmodem241201"`.

The command line we enter into the terminal should then look like this:

```
stty -f /dev/tty.usbmodem241201 1200; sleep 3; avrdude -v -p atmega32u4 -c  
avr109 -P /dev/cu.usbmodem241201 -b57600 -D -Uflash:w:"firmware.hex":i
```

Copy this line into the Terminal window and execute it.

After approx. 5 seconds the firmware was then transferred

This example should only show how it basically works, and which ways there are. Further information and instructions can be found on numerous websites on the Internet.

If the transfer of the firmware via USB does not succeed, the bootloader of the Pro Micro must be reset. This is rarely necessary, but it is always recommended to be on the safe side and to reset the Arduino ProMicro to its factory defaults so that the firmware can be transferred via USB again.

VCC MOSI GND

2 4 6

1 3 5

0

MISO SCK RST

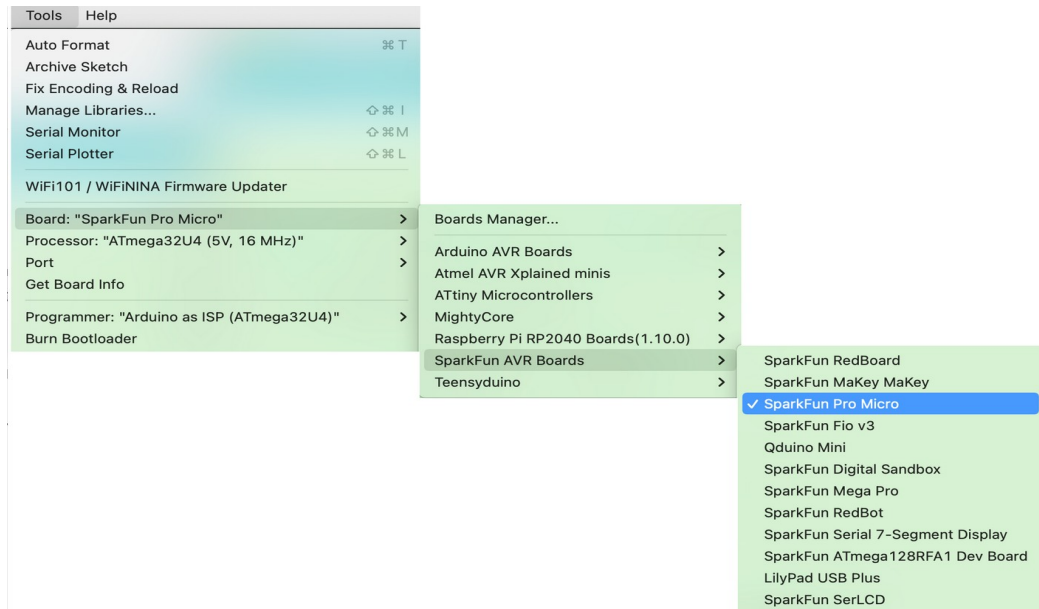


(you need one of the programmers in the right list)



The process itself takes place via the installed **Arduino IDE**

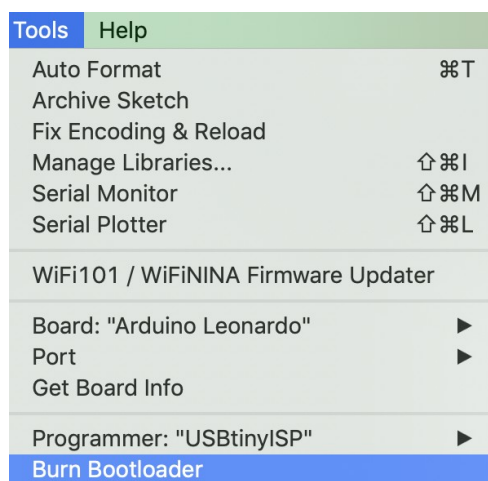
To do this, the right device must first be selected:



(if not available, see the next page)

After that, the bootloader is transferred via the menu item

„**Tools**“ → Burn Bootloader:



If the SparkFun item does not exist, the board support must be installed first:

