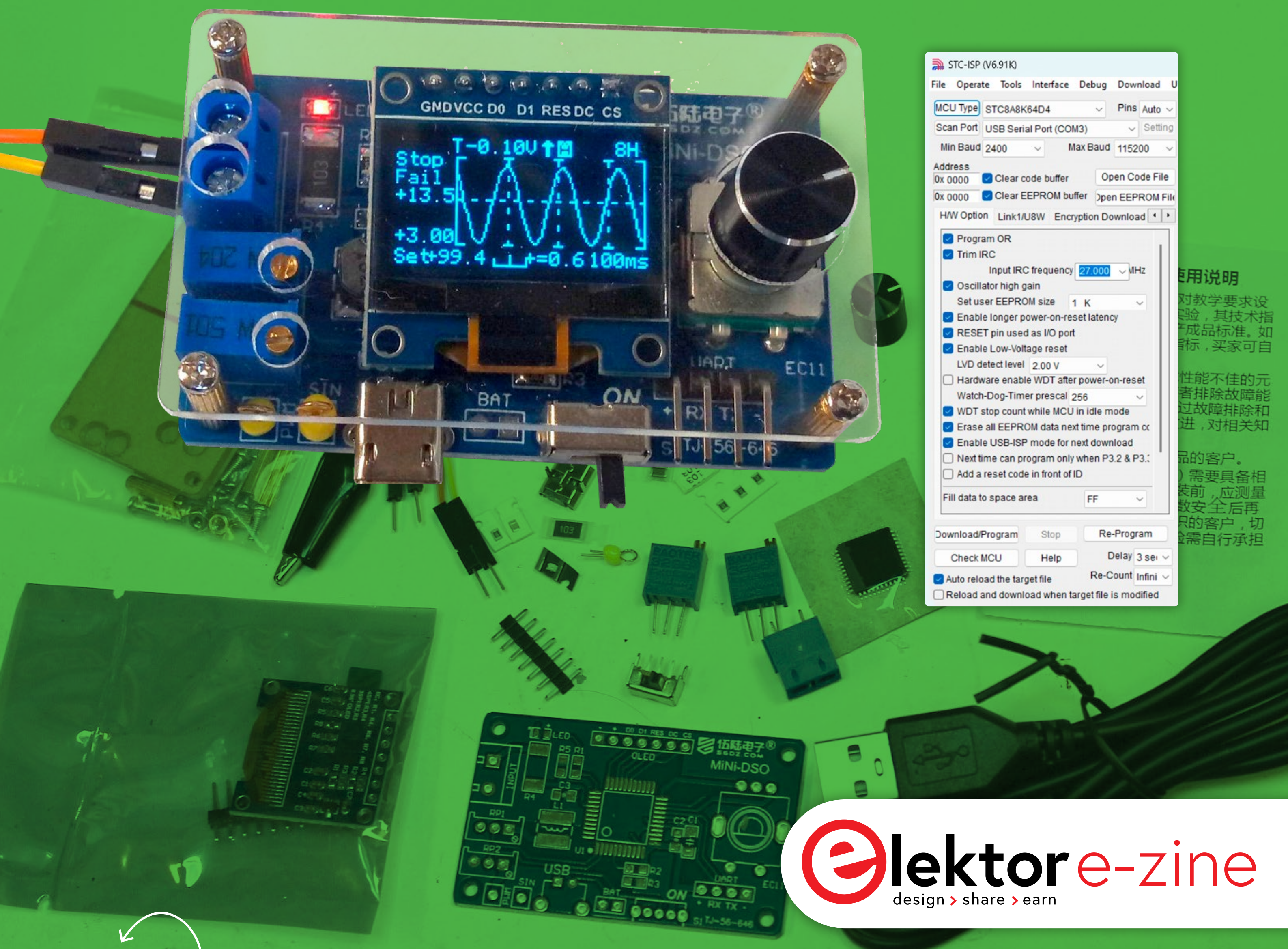
# reverse project #03

an e-zine exclusive

August 2024

# HACKING A MINI OLED OSCILLOSCOPE KIT

## Making It Open Source Again




elektor e-zine
design > share > earn

SUBSCRIBE NOW AT
www.elektormagazine.com/elektor-newsletter

reverse project e-zine Exclusive


*Figure 1: The contents of the Mini OLED Oscilloscope kit. Will this become my new workhorse?*

Elektor lab · Elektor lab · TESTED · Elektor lab · Elektor lab

# Hacking a Mini OLED Oscilloscope Kit

## Making It Open Source Again

**By Clemens Valens (Elektor)**

The Mini OLED Oscilloscope is an inexpensive kit to build a tiny digital oscilloscope with. Besides the power switch, it has only one control, a rotary encoder with a built-in pushbutton. The kit's microcontroller comes preprogrammed, and the software is not available. Hmm. Let's see what we can do about that.

The Mini OLED Oscilloscope is tiny indeed, as it measures only 57 mm by 38 mm. Its height, when put inside its "enclosure" (two small transparent acrylic plates), is 26 mm. Tiny also means that many components are SMT devices, including the 44-pin microcontroller (**Figure 1**). Therefore, assembling the kit means soldering SMT parts, a good exercise.

The display is a small OLED display with a resolution of 128 by 64 pixels. The scope features one channel that can measure signals up to 100 kHz. The maximum input voltage is 30 V, the minimum voltage is 0 V.

### Specifications

- Vertical range: 0 to 30 V
- Horizontal range: 100 µs to 500 ms
- Trigger type: auto, normal and single
- Trigger edge: rising and falling
- Trigger level: 0 to 30 V
- Run/Stop mode
- Automatic frequency measurement
- Power: 5 V micro-USB
- 10 Hz, 5 V sinewave output
- 9 kHz, 0 to 4.8 V square wave output

## Password-Protected Assembly Manual

An assembly manual is not included in the kit, only a tiny piece of paper with Chinese print on it and a QR code. The QR code takes you to a webpage in Chinese [1]. Once you manage to translate it, you'll discover that a password is required to download the product's documentation and other supporting files. If you read Chinese, you already knew this, as it is printed below the QR code (duh). To save you the trouble, the password is "DFRE".

All the files inside the archive that you download this way have Chinese names and their contents are in Chinese, but with the help of ChatGPT you can work your way through it. I found that assembling the kit was easy when referring to the schematic. This is possible because the oscilloscope is not much more than a microcontroller with a resistor divider on an analog input together with a few decoupling capacitors (see **Figure 2**). Therefore, assembling the printed circuit board takes not more than half an hour. You must also assemble the test lead yourself (it is a kit after all). It consists of two crocodile clips on 20-cm-long Dupont wires.

Once everything is assembled, you can try it out. Connect a 5 V phone charger with micro-USB connector to the device and switch it on. The test lead connects to the screw terminal on the left side of the screen.

*Figure 2: Can a digital oscilloscope be this simple?*

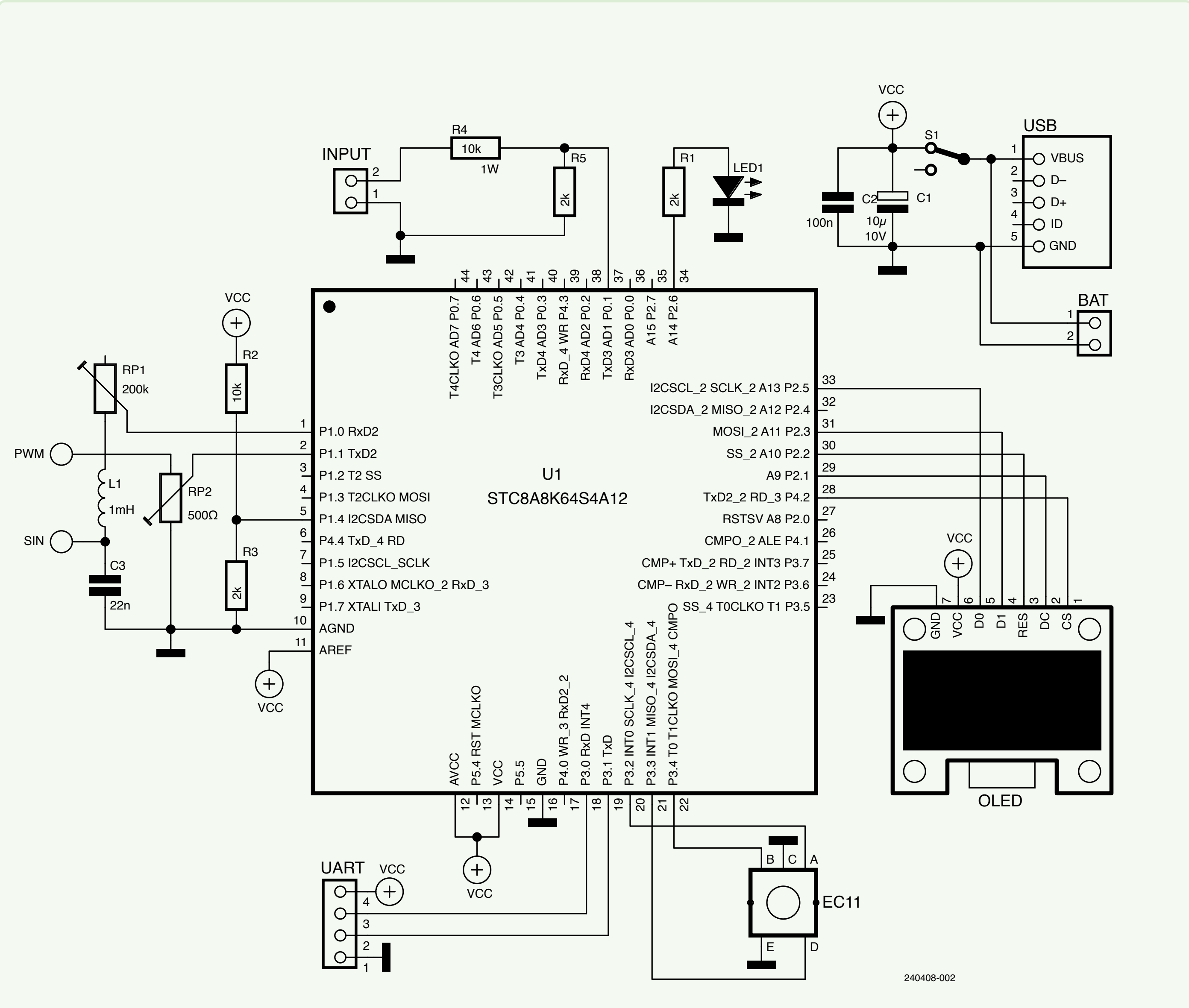


*Figure 3: Use the STC-ISP tool to get some information about the MCU.*

```
Checking target MCU ...
  MCU type: STC8A8K64D4
  F/W version: 7.4.3U

Current H/W Option:
  . ISP-IRC frequency: 24.038MHz
  . IRC frequency: 27.026MHz
  . Wakeup Timer frequency: 36.025KHz
  . Oscillator gain is HIGH
  . User EEPROM size is 1 K
  . Do not detect the level of P3.2 and P3.3 next download
  . Power-on reset, use the extra power-on delay
  . RESET pin behaves as IO pin
  . Reset while detect a Low-Voltage
  . Thresh voltage level of the built-in LVD : 2.00 V
  . Hardware do not enable Watch-Dog-Timer
  . Watch-Dog-Timer pre-scalar : 256
  . Watch-Dog-Timer stop count in idle mode
  . Erase user EEPROM area at next download
  . Do not control 485 at next download
  . Do not check user password next download
  . Reference voltage: 1187 mV (Range: 1100~1300mV)
  . Testing time: 2024-1-2

  MCU type: STC8A8K64D4
  F/W version: 7.4.3U


  Complete !(2024-08-09 09:57:57)
```

## Moving Around the Display

When powered on, the device shows a DSO-like screen with a 2 by 3.7 grid (the cells on the right don't fit completely on the screen) surrounded by values. These are the oscilloscope settings.

In the bottom-right corner is the horizontal time scale, and you can adjust it with the rotary controller. To move to another parameter, you must turn the encoder while keeping it pressed down.

The lower-left corner shows *Set*. When you select it and then turn the encoder without pressing it, a setup menu opens. Here you can choose between dot and vector display, change the display brightness and adjust a mysterious parameter called *LSB*.

## Auto-Ranging

The vertical range (middle left) parameter has a surprise. If you keep turning the controller to the right, beyond the minimum range value, it switches to auto-ranging. This is useful for finding the input signal in case it isn't showing. Turning back to the left returns to manual ranging.

While fiddling around with the rotary encoder, you will quickly notice that it isn't the best controller in the world. Well, the encoder is fine, but the software using it, isn't. In particular, parameters that can take on only two values (e.g. trigger edge) are very complicated to adjust. The push-and-spin way of moving around the screen isn't easy either and keeps toggling the DSO between Run and Stop mode. It is complicated to get a signal on the screen, but, as said before, using the auto-ranging option can help you out here.

## Extra Functions

Why is not completely clear to me, but the Mini OLED Oscilloscope has a 10 Hz, 5 V sinewave output. It is obtained by lowpass filtering a 9 kHz PWM signal. Trim potentiometer RP2 is involved too but doesn't seem to do anything unless you set it close to its minimum value. Then the sinewave becomes ugly. Consulting the schematic, you'll find that RP2 adjusts the cut-off frequency of the low-pass filter. Therefore, setting it too high, it won't filter very well, and the output becomes an interesting test signal.

There is also a PWM output, but in practice there is only a 9 kHz square wave available. Its amplitude can be adjusted with RP1.

Finally, there is a serial port. It is not used by the oscilloscope, but it can be used to upload new firmware to the microcontroller.

## Back to the Roots

The archive with the assembly instructions does not include source code nor any precompiled binaries. However, when I googled the title of the schematic (*STC8A8K-MiniDSO*), I immediately found what seems to be the origin of the Mini OLED Oscilloscope. Apparently, the original design is by Creative Lau [2]. His repository entitled Mini-DSO has a schematic that is very similar to the one shown in this article. Better yet is that it contains the source code and a link to a video showing the device in action.

Watching the video [3] is instructive because not only does it provide information on how to use the Mini DSO (and what the *LSB* parameter mentioned before is for: calibrating the input voltage) but also how the design was intended to be used. Once you know that it is supposed to be a hand-held pen-type oscilloscope that can be controlled with only your thumb, certain design choices start to make sense. Unfortunately, the designer of the Mini OLED Oscilloscope kit apparently did not watch this video and turned the device into something rather impractical and useless.

## A Hand-Held Oscilloscope

The probe is connected to the input on the left because it should be a metal rod (a pen), not wires with crocodile clips. Second, the oscilloscope is controlled with only your thumb by turning and pressing the rotary encoder. This is simply impossible with the small, smooth knob included in the kit. Creative Lau used a different knob for a reason.

Being intended as a hand-held instrument, the power supply is supposed to be a battery, not a USB phone charger. Battery power is possible because a battery connector can be mounted, but the voltage regulator that turned the battery voltage into a nice and stable 5 V was left out of the kit.

## A Better User Interface

The user interface of the Mini-DSO shown in the video differs slightly from the one on the Mini OLED Oscilloscope. The Mini-DSO looks a bit nicer. Therefore, the question arose: would it be possible to put the Mini-DSO software on the Mini OLED Oscilloscope? The circuits are almost identical.

Both oscilloscopes are based on an STC8A8K64S4A12 from STCmicro (a name that sounds strangely familiar). This is one of those super-cheap, super-fast 8051 clones. It is a so-called 1T device, meaning that an instruction is executed in a single clock cycle. The STC8A8K64S4A12 is the flagship of the STC8F family, boasting 63 KB flash memory, 1 KB EEPROM, and 8 KB RAM. It can be programmed in-circuit over a serial port without requiring any special tools. The STC-ISP tool ([4], an amazing piece of work, BTW) that lets you do this can be downloaded for free from the STCmicro website.

## Porting the Firmware to SDCC

The complicated part of porting the Mini-DSO software to the Mini OLED Oscilloscope is the fact that it is in the shape of a project for Keil's C51 compiler. This compiler is everything but free, and the evaluation version won't create the HEX file as the project is too big. A way around this is using the open-source compiler SDCC instead. Unfortunately, SDCC doesn't use the same syntax for defining processor-specific things like special function registers (SFRs) and addressable bits, which means going through the code and changing everything. Also, a header file with register definitions for the STC8A8K64S4A12 is required and, of course, a make file to compile everything and link it together. Another project that I found on GitHub, a generic support library for some of the STC8 MCUs [5], turned out to be really helpful here as it is compatible with both C51 and SDCC. This let me create a Makefile that worked, and a usable processor register definition file.

## In-Circuit Programming

After a few hours of searching, comparing, and editing, I managed to get SDCC to compile the Mini-DSO project without warnings and errors. I adapted the MCU pin mapping from the Mini-DSO to the Mini OLED Oscilloscope (pin definitions are in the file *config_stc.h*). Then I connected a USB-to-serial converter to the serial port of the device and fired up the STC-ISP tool.

I switched off the device, clicked the *Check MCU* button and switched the device on again. Information about the MCU appeared in the output window (**Figure 3**). An interesting observation here was that it showed the MCU as an STC8A8K64*D4* instead of an *S4A12*. Note how I used STC-ISP version V6.91K. It proposes to download a newer version (V6.94K), which I did, but this version doesn't seem to have the *Check MCU* button. At least, I couldn't find it.

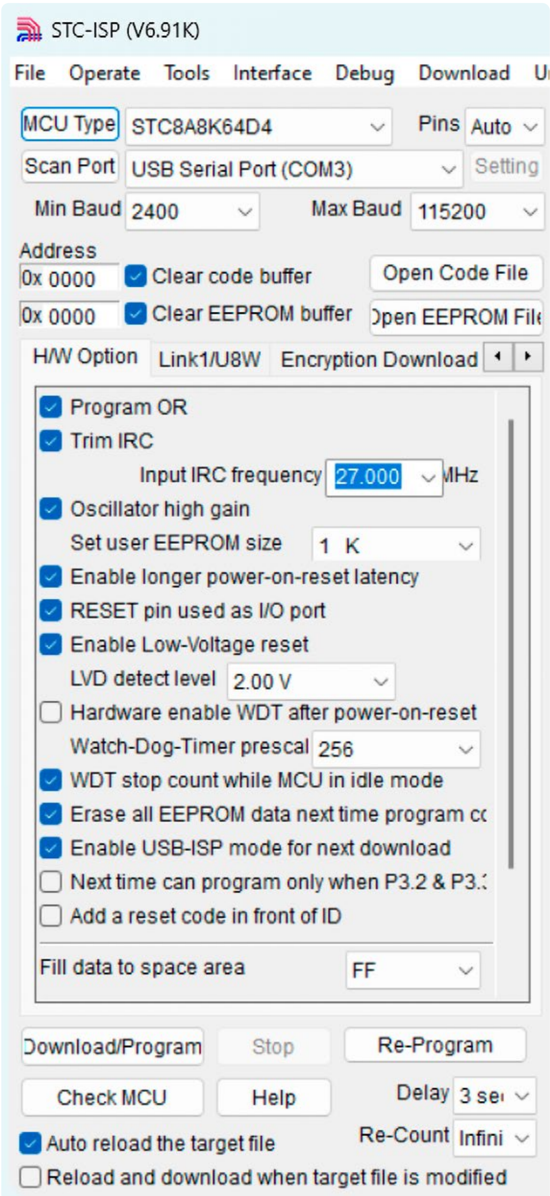
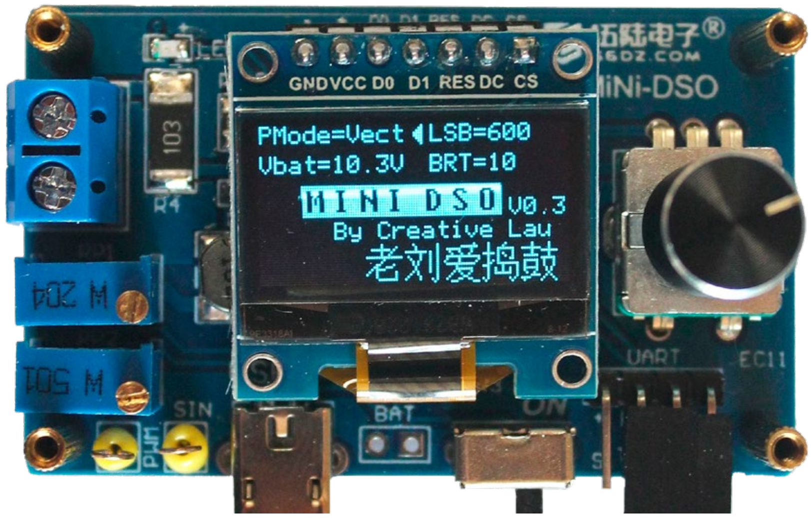



*Figure 4: These are the fuse settings that I used and that seem to work.*




▲

*Figure 5: Mission completed: Mini-DSO has replaced Mini OLED Oscilloscope.*

I then proceeded to upload my new HEX file. This is easy. Again, first switch of the board, load the HEX file in the STC-ISP tool and click the *Download/Program* button, then switch on the board to start programming. It takes about ten seconds to complete. And what do you know? Indeed, it didn't work. And even though it was more or less what I expected, I still felt a bit disappointed.

## Debug Some Issues

Now there are many places where porting software can go wrong. Possible suspects are the programming options, which I had left at their default values. The MCU check listed values for these options, so these may be a kind of configuration fuse settings. Running the MCU check once more showed that two values had changed: EEPROM size and the IRC frequency. After uploading the program a second time, but now with the changed MCU configuration settings set to the values I'd read the first time (**Figure 4**), the LED suddenly came alive. Even better, turning the encoder influenced the behaviour of the LED. As Creative Lau explains in his video, this LED shows sampling status, meaning that the program seemed to be running! Hope. So maybe now the only thing not working was the display.

Poking around on the OLED pins with a real oscilloscope revealed lots of activity on the DO and D1 pins but nothing on CS and DC. The source code shows that these pins are used for synchronization (the OLED display expects a 4-pin SPI protocol), so there should be some activity. Closer inspection with the real oscilloscope suggested that the pins were not configured properly, as they did show some weak activity. Indeed, in the function main, the new I/O pins were not activated. After correcting this, the display started working too. It wasn't perfect, but at least it showed that the software was running. More importantly, it showed that I could now use SDCC to write new software for the little board and execute it. From this point on, it was just a matter of time to get the Mini-DSO working.

## Cracked It!

To make a long story short, after some more hours of debugging, I finally discovered that the problem was SDCC not producing proper code for reading a string from a table in program memory. The Mini-DSO uses two of these tables, and reading them the SDCC way makes the program crash. Replacing the tables by functions fixed the issue. Now, not only does the Mini-DSO program run on the Mini OLED Oscilloscope hardware (**Figure 5**), but it can also be compiled with open-source tools.

The Mini-DSO software ported to SDCC can be found at Elektor Labs [6]. Enjoy!

240408-01

## Questions or Comments?

Do you have technical questions or comments about his article? Email the author at clemens.valens@elektor.com or contact Elektor at editor@elektor.com.

## Related Products

- **DIY Mini Digital Oscilloscope Kit**
  www.elektor.com/20938
- **C. Valens, *Mastering Microcontrollers Helped by Arduino* (Elektor, 2017)**
  www.elektor.com/17967

## WEB LINKS


[1] Mini OLED Oscilloscope kit: https://www.56dz.com
[2] Mini-DSO by Creative Lau (GitHub): https://github.com/CreativeLau/Mini-DSO
[3] Video by Creative Lau: https://www.youtube.com/watch?v=-8PadlS7c4c
[4] STC-ISP tool: https://www.stcmicro.com/rar/stc-isp-en.rar
[5] FwLib_STC8, a generic library for the STC8G/STC8H MCU series (GitHub): https://github.com/IOsetting/FwLib_STC8
[6] Downloads for this article: https://www.elektormagazine.com/labs/reverse-project-mini-oscilloscope

# Join the Elektor Community

## Take out a GOLD membership!

- The Elektor web archive from 1974!
- 8x Elektor Magazine (print)
- 8x Elektor Magazine (digital)
- 10% discount in our web shop and exclusive offers
- Access to more than 5,000 Gerber files

## Also available

The Digital GREEN membership!

- The Elektor web archive from 1974!
- 8x Elektor Magazine (digital)
- 10% discount in our web shop and exclusive offers
- Access to more than 5,000 Gerber files

**www.elektormagazine.com/member**

elektor design > share > earn