Work Diary-MinioscilIoscope

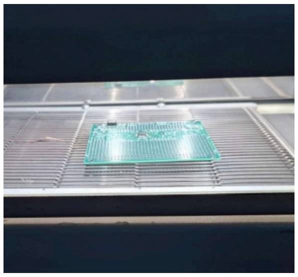
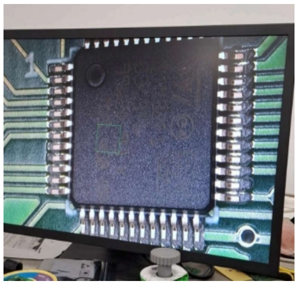
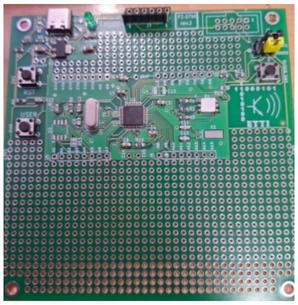
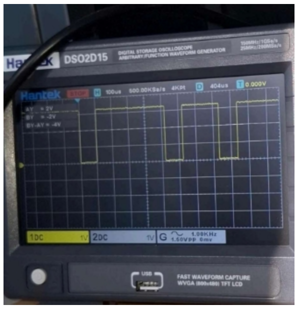
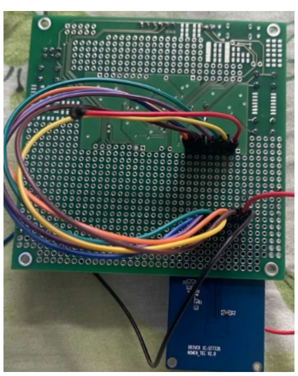
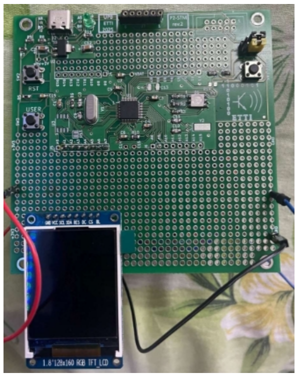
1.Description

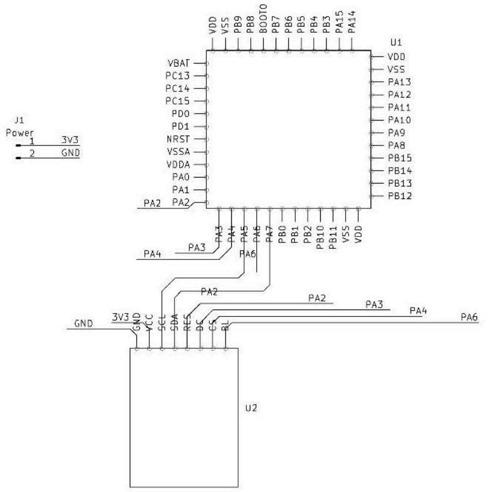
The chosen topic was the realization of a mini-oscilloscope using a STM32 microcontroller, a ST7735S TFT display, and the microcontroller's internal analog-to-digital converter (ADC). The main purpose of the project is to take an analog signal (from an external signal generator), convert it to digital using the ADC, and display the waveform in real time on the display screen.

To address the overall theme, we made the following implementation decisions. The ST7735S display was chosen because of its resolution (128x160 pixels). The internal ADC a1 of the STM32 microcontroller is used to read the signal. The display is done by plotting the points corresponding to the ADC values on the vertical axis of the display, moving horizontally. Thus, the signal is rendered graphically, similar to a real oscilloscope.

The operating logic is as follows. The signal generator produces an analog signal that we want to measure. The analog signal goes to the input of the microcontroller's ADC (Analog-to-Digital Converter). The ADC-u1 converts the analog voltage to a digital numeric (12-bit) value.
The microcontroller takes quick readings of the digital values from the ADC at regular intervals (sampling rate). This step is crucial to capture the waveform as accurately as possible. The read digital values are temporarily stored in a buffer (a vector) so that they can be processed and displayed. The processed values are transformed into X,Y coordinates on the ST7735S display. A continuous loop is made: read ADC -+ process -+ display -' read ADC -+ ... so that the signal is displayed in real time, like a classic oscilloscope.

2. Hardware description

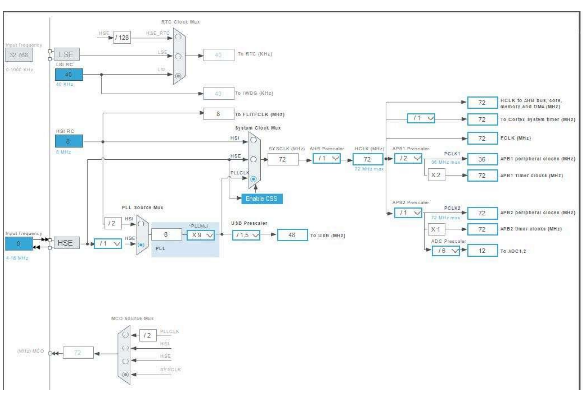







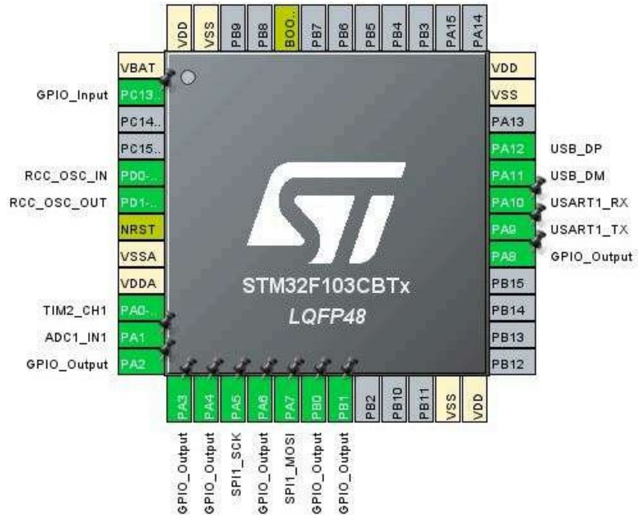


Cantitate	Piesa	Pret (RON)
1	PCB Facultate (STM32F103CBT6)	-
1	LCD TFT 1.8' ST7735S	35
8	fire mama-tata	
2	fire tata-tata	
1	USB-TTL	8

3. Software description

We used SPI ports for communication with the disp1ay-u1 ST7735S (e.g. MOSI, SCK, CS, DC, RESET), analog port for ADC (ADC1_IN1). We used a single ADC channel for signal capture.
Reference voltage is minimum 2.4V , maximum VDDA (2.4-3.6[V]). Presealer-u1 ADC is 6, providing ADC frequency of 12 MHz (maximum allowed is 14 MHz, minimum 0.6 MHz). The resolution of the ADC is 12 bits. The sampling frequency is minimum 0.05 MHz and maximum 1 MHz.






```
arm-none-eabi-size build/Test1.elf
   text    data     bss     dec     hex filename
  35116     480    7088   42684    a6bc build/Test1.elf
arm-none-eabi-objcopy -O ihex build/Test1.elf build/Test1.hex
arm-none-eabi-objcopy -O binary -S build/Test1.elf build/Test1.bin
```

STM32 Memory Usage Report

		Size
1	.text (code)	35116 bytes
2	.data (initialized vars)	480 bytes
3	.bss (uninitialized vars)	7088 bytes
4	Total .bin (text+data)	35596 bytes

- **FLASH**: 35.6 KB din 128 KB (≈ 27.16%)
- **RAM**: 7.57 KB din 20 KB (≈ 36.95%)

4. Results and conclusions

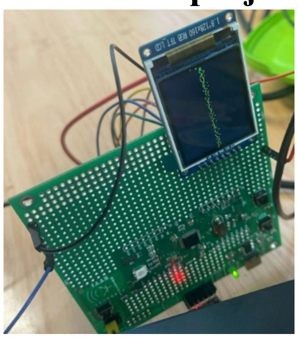
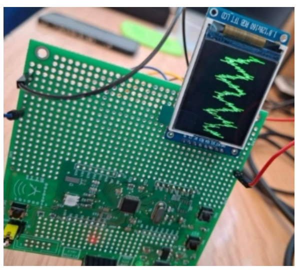
The initial requirements of the project were relatively simple and straightforward: realization of a mini-oscilloscope using the microcontroller's internal ADC-u1 and waveform display on a graphical display of choice (e.g. ST7735S). During the development of the project, a number of practical and optimization challenges arose, which led to adjustments of the initial implementation.

Initially, the displayed signal was quite erratic and noisy, due to the nature of the actual analog signal combined with the lack of a hardware filter and the limitations of the internal ADC. To mitigate this effect, we implemented an averaging filter in software. Ann averaged between three consecutive samples (i-1, i, i+1) which led to a noticeable improvement in the visual stability of the waveform, bar bar High performance posts. Displaying a single pixel for each "filtered" value provided a good compromise between clarity and rendering speed.

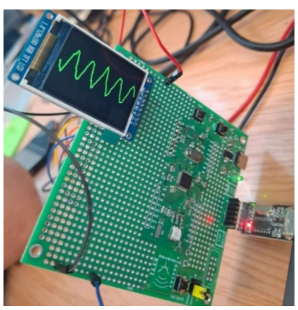
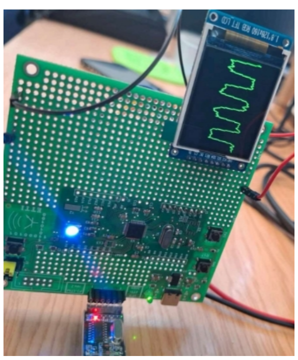
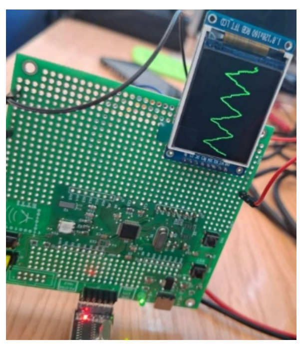
Also, the ST7735S display was configured in portrait mode, which limited the signal length on the X-axis. We decided to change the orientation to landscape mode, so that the time (X) axis would benefit from more pixels and the waveform would be represented more intuitively.

One of the significant challenges encountered during project development was related to the unipolar nature of the STM32 microcontroller's internal a1 ADC. It can only measure positive voltages, between 0V and Vref(in our case 3.3V). Therefore, if a bipolar signal, such as a sinusoidal signal with symmetric variation around 0V (e.g. -1.5V ... +1.5V), was directly applied, it would exceed the input specifications of the ADC, risking permanent damage to the input pin or even the microcontroller. To make the signal compatible with the unipolar a1 domain of the ADC, we applied a simple but effective solution: adding a DC offset to the analog signal at the input, so that the negative part is translated into the positive domain. This problem can also be fixed with a hardware solution. In order for the signal to be displayed correctly and centered on the horizontal axis (virtual 0V), the offset introduced in hardware was eliminated in software. Specifically, when processing the data read from the ADC, the offset was subtracted from each sample.

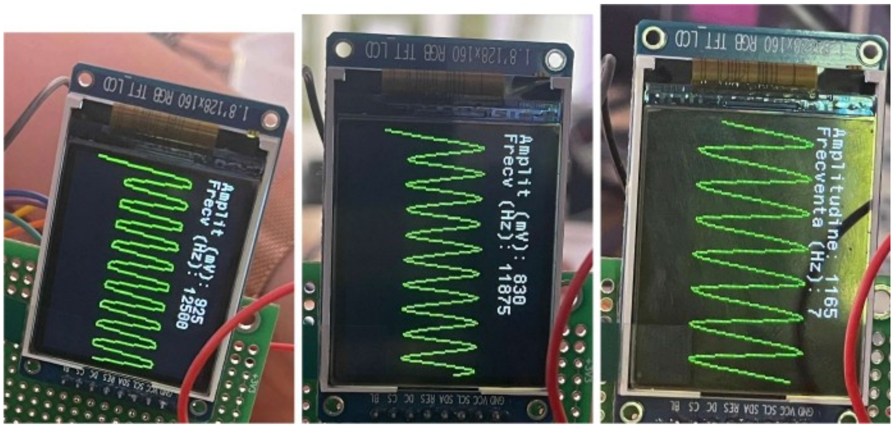
First phase of the project

Second phase of the project

Final phase of the project



5. Bibliography

- STM32F103CBT6 datasheet - available on [datasheetstm com]
- Similar projects_
- ST7735 screen tutorial ______

S 2025