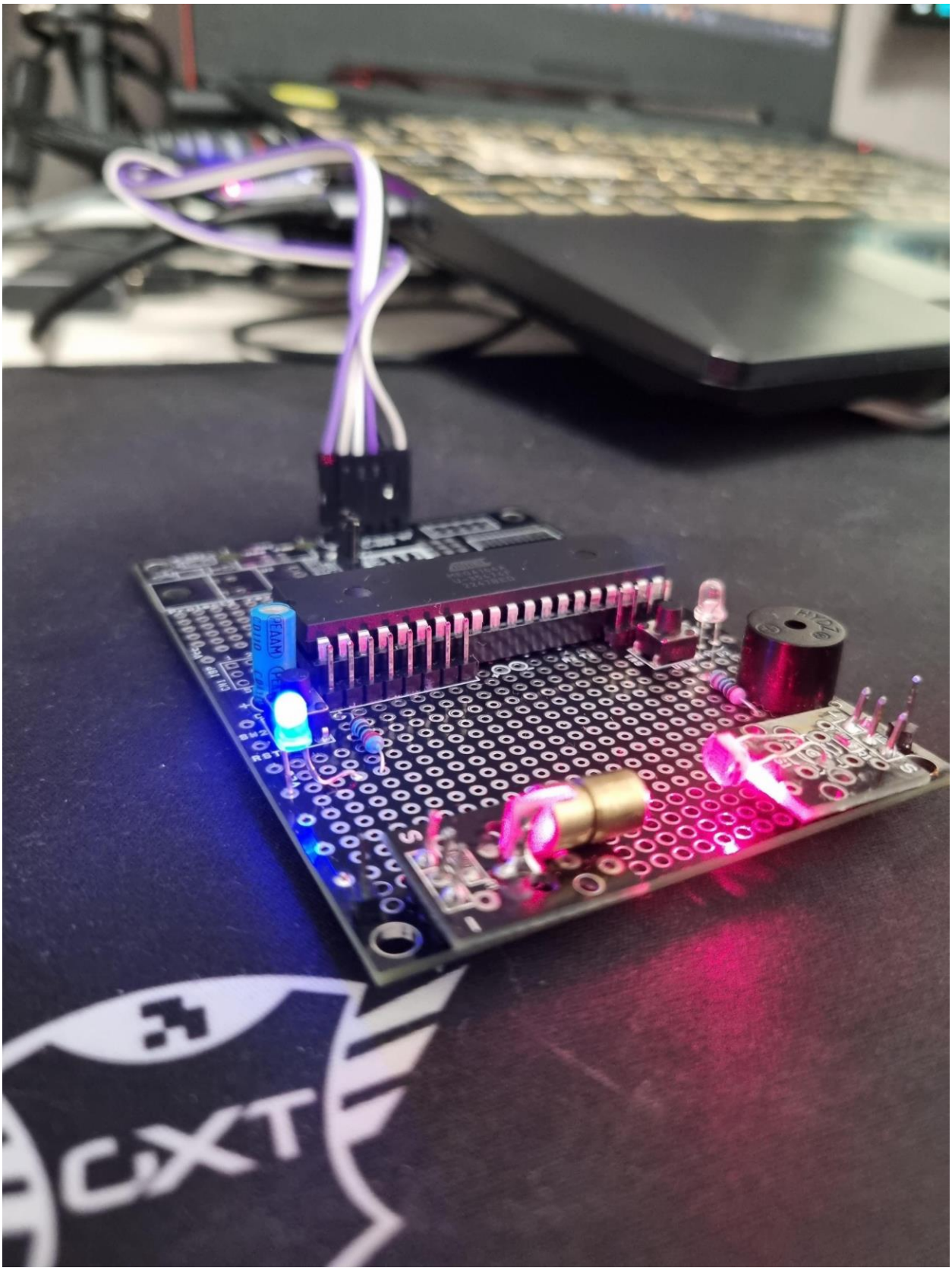

CXT

Security system using laser sensors

Barău Elena-Raluca Mărculescu Tudor | ETTI 431F |

project coordinator: Conf. Dr. Ing. Bogdan-Cristian Florea

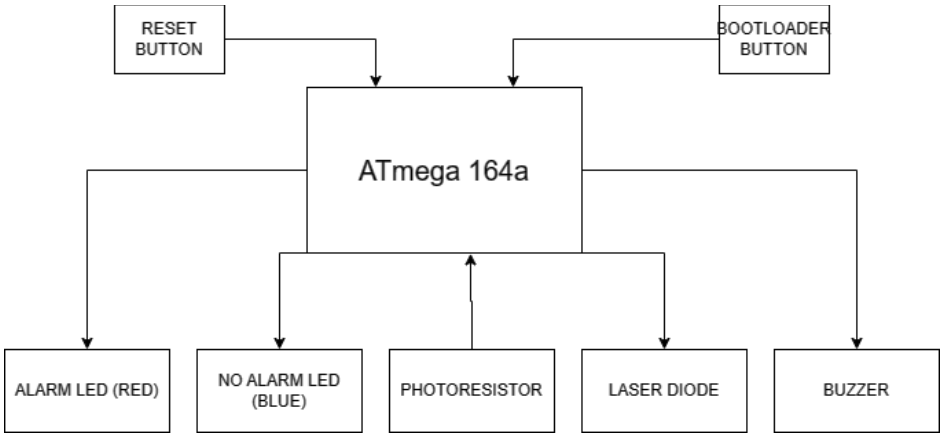
Project requirements

This project consists in implementing a security system which detects the position of a door using a laser diode and a photoresistor. When the door opens and the laser beam no longer falls on the sensor, it signals a visual and audio alarm using a red led and an active buzzer.

The system can be implemented in two ways. One way is having one sensor on the door and the laser on the alarm module. The other way would be putting the sensor and alarm module next to the door.

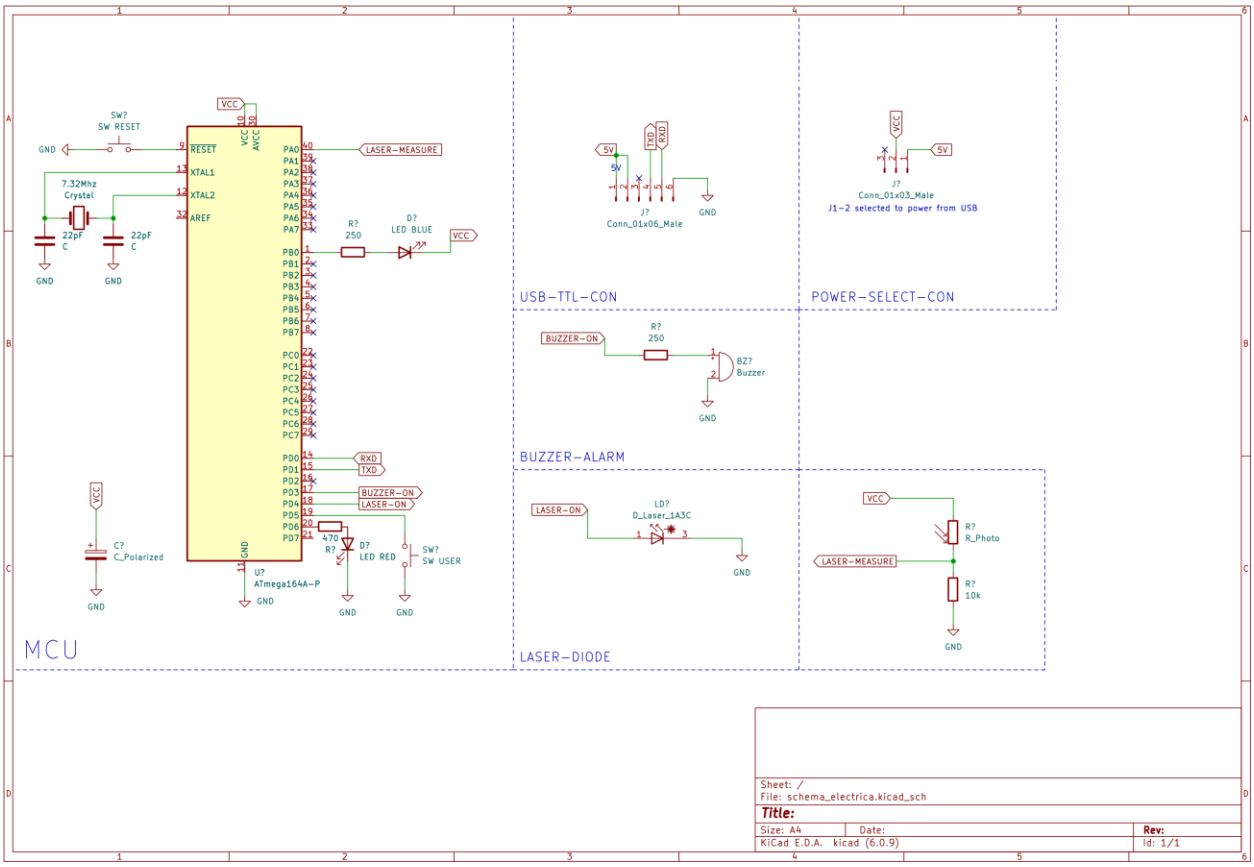
The expected results are for the system to ring the buzzer when an object is detected in between the laser light and the photoresistor.

Block diagram



The hearth of the system is the ATmega 164a microcontroller. It controls the photoresistor, the laser diode as well as the buzzer and the LEDs. The system is made such that when pressing the reset button, the microcontroller starts from a *"no alarm"* state. When an alarm starts, the user needs to press the reset button in order to stop it.

Electric schematic

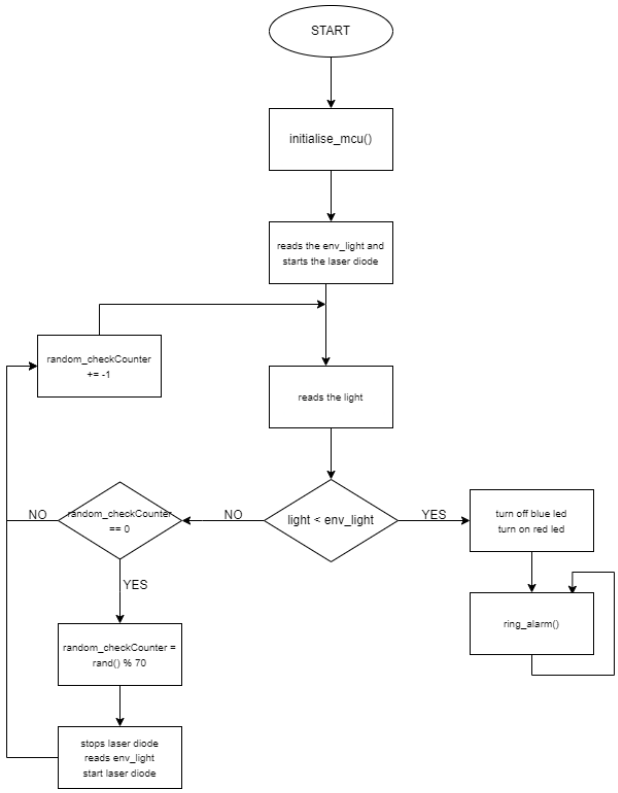


The schematic was plotted in KiCAD. The important aspects is the use of PD3 as well as PD4 ports as digital outputs, in order to control the buzzer as well as the laser diode. PA0 pin is used in order to measure via the built-in ADC the voltage drop across the 10k resistor in the photoresistor sensor module.

The project powers up via the USB-TTL converter that is connected to a laptop. It has a 5V input voltage from the VCC rail of the USB.

The built in ADC has a 10-bit resolution, so the expected output values are in the range of 0-1023. It is configured to the VCC voltage reference. It is also set to be Free Running such that it can convert the read voltage values faster. For our project is important to have a fast conversion rate.

Firmware diagram



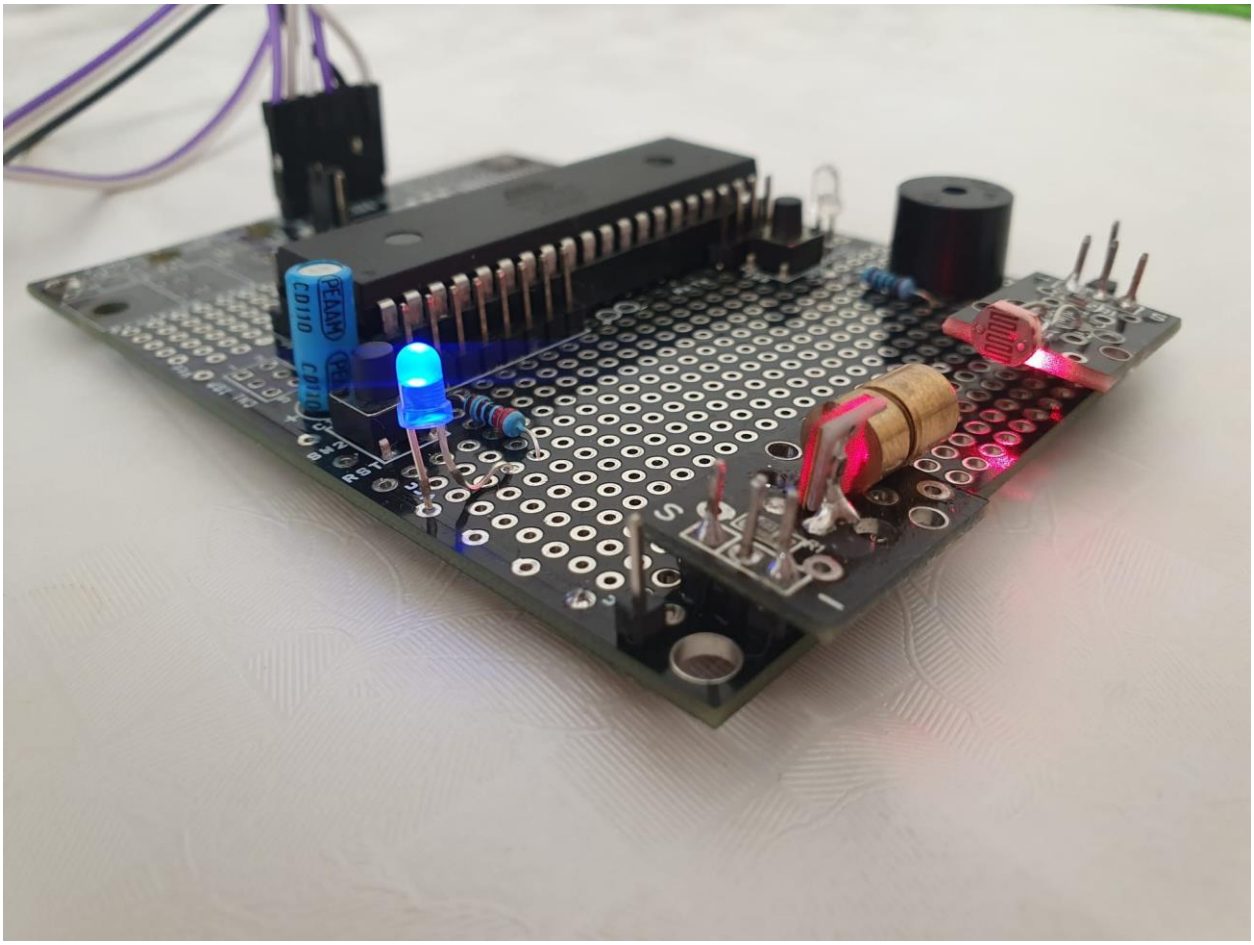
The firmware of the device was kept as simple as possible in order to minimize errors. In the initialization phase, all the pins are set to the corresponding values of input / output and the ADC is set to Free Running. The laser is not yet started. Afterwards, the device reads the light in the environment and then starts the laser in order to measure again the light with the laser shining on it.

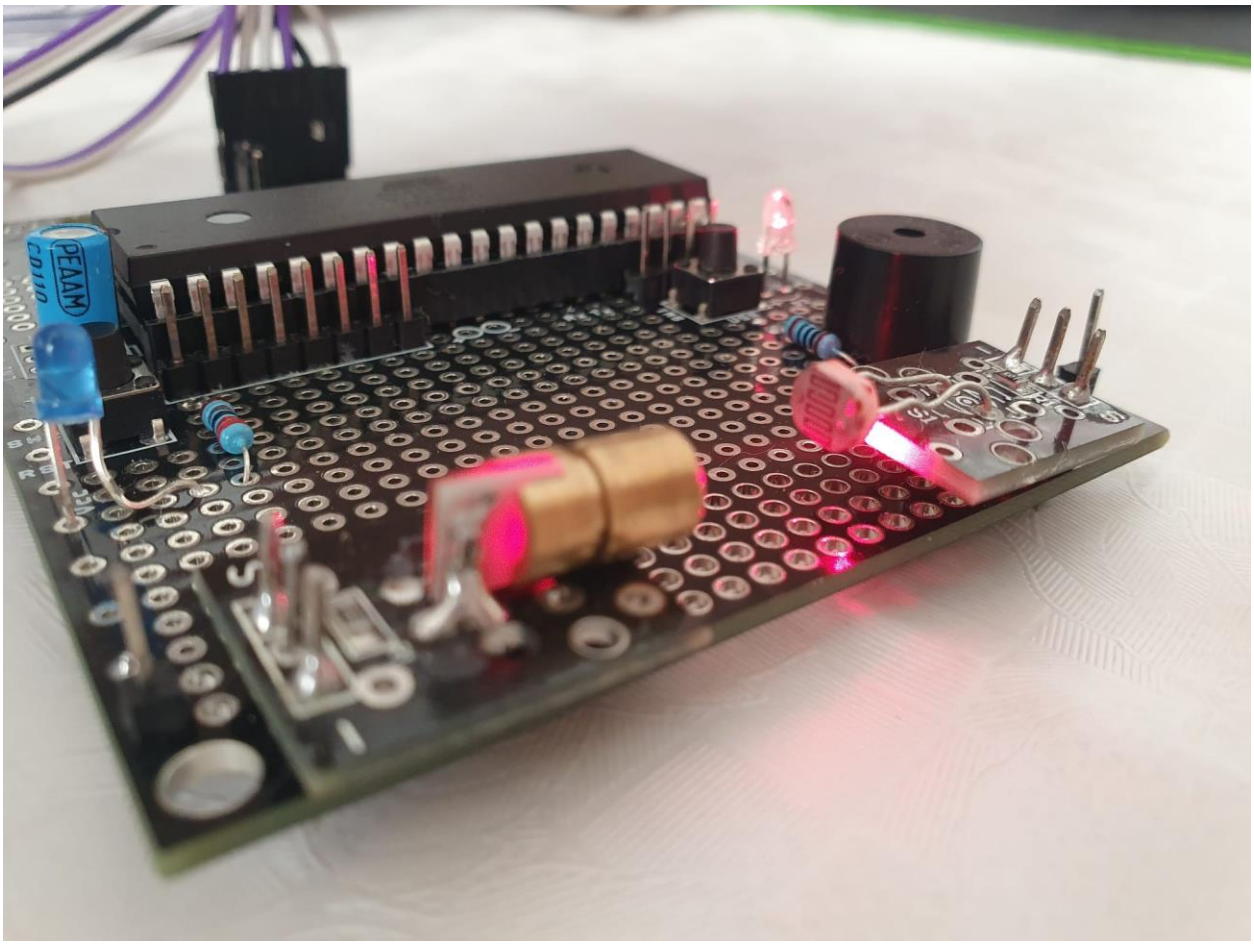
When the brightness of the light in the environment is higher than the brightness of the light in the environment + the brightness of the laser, a state of alarm is issued. The red LED is turned on and the buzzer will start ringing.

Because the system reads the brightness of the environment light randomly within 30 seconds, it makes it hard for a possible attacker to confuse the system by making it read a fixed amount of brightness that is just right in order to make the system think that the door has not opened even though it had. The randomness factor is an important part of any security system.

Pictures of the project

Alarm off





*1 - **Alarm on***

Personal conclusions

1. The system works good in any environment from the tests that we ran until now.
2. It is a simple device that might be a good choice for someone that wants a cheap solution.
3. Programming a microcontroller without a debugger is harder because you can not see the state of memory or registers at any given moment of time.

Problems and solutions

- Problem: The ADC read only the maximum value of 1024. **The solution** was to set it to Free Running and to set the reference to AVCC, not the internal 2.56V one.
- Problem: The ADC read abnormally high values for the brightness of light in the environment. **The solution** was to not shine the laser directly on the photoresistor and delay the reading by 100ms from the moment of turning off the laser beam.

- Problem: Could not communicate with the bootloader. **The solution** was to connect the uTX pin to TX pin on the USB-TTL converter as well as uRX pin to RX pin.

References

1. Documentatie pentru proiectul 2 Informatii pt studentii la Proiectul 2/3. (pub.ro) (accesat ultima data in data de 28.05.2023)
2. Explicatii si comentarii pe softul de test Microsoft Word - test soft description-2018 (pub.ro) (accesat ultima data in 28.05.2023)
3. ATmega164a datasheet ATmega164A (pub.ro) (accesat ultima data in 28.05.2023)
4. How to Program AVR Microcontroller Atmega16 Using USBASP programmer and Atmel Studio 7.0 (circuitdigest.com) (accesat ultima data in 01.03.2023)
5. Tutorial interfatare intre senzori si microcontoler Microsoft PowerPoint - 04-interfacing.ppt (pub.ro) (accesat ultima data in data de 1.05.2023)

Annex

Source code, Firmware-security.zip, Firmware-security.zip