Technical description of chronometer project

Table of Contents

1. INTRODUCTION:

As an introduction, we would like to highlight the purpose of this report, which is more or less to put in writing all the detailed information, all the steps carried out together with all the errors and issues that we face and how we have solved them when attending the lab class we had so far. The subject of our project as mentioned above is to create a timeline from scratch. Content of the report will be based on the hardware side: description, methods, parameters, features. On the software side: we will explain the parameters and functionalities of the written program, and of course the issues encountered throughout.

2. ASSUMPTIONS:

On the table below [Table 1] we can see the assumptions

Functional Assumptions	***Design Assumptions***
Aim of our project is to create a device so called chronometer	Main steps for an successful accomplishment of this project:
Chronometer can tell is how fast the bullet was traveling when it passed by the tube	Wiring the Arduino with all the required components to be able to connect and display the needed message for the user on the screen for a start
It lets you start mapping out the expected flight path of the projectile	Connection and ignition of the display after programming correctly the Arduino
The speed the bullet leaves the barrel, is expressed in M/S	Testing the result for errors checking before assembly
Velocity is shown on the screen of our device	Painting the box to be black
Device with relatively good accuracy	Testing many Tubes for the best shots accuracy
This information then lets us adjust our aim point (through the turrets or holding over) to get that bullet to land where we want it to.	Changing guns and bullets to obtain precise results
If you are reloading, then you may or may not have an indication of the expected velocity	Small changes in the source code
A chronograph is the best method of determining accurate information to use in such situations calculations	Finally designed whole device, put into the box and assemble the components for final presentation

Table 1

3. HARDWARE DESCRIPTION:

3.1 Description and block diagram:

- The MCU chosen for this project was the Arduino UNO because it is sufficiently fast and relatively cheap. The main advantage is the easy way of programming and high capability of components.
- Sensors used were the infrared light sensors TCRT5000 LM393, they are quite precise and easy to mount and rearrange For the user interface we used an LCD screen connected to the MCU with a I2C standard connection and 2 LED's.
- This interface was chosen because I2C interface is much easier to connect and configure instead of serial connection to the Arduino.
- Box is KRADEX Z50B – dimensions 147x92x50mm, black painted. Sufficiently big
- 2 simple 5mm diameter LEDs for indicating the status.
- Hardware was the hardest part in the project. It contained many researches in terms like screen choosing, pipe choice and sensor assembly. the 9V battery as a power source is connected to an ON/OFF switch, directly connected to Arduino. Screen we have chosen is 16x2 LCD screen with I2C interface, which is sufficiently large to display all necessary information. Two sensors TCRT5000 LM393 IR are remoulded.
- The emitting diode has been unsoldered and pointed on the receiving diode, creating the photocell. Bullet between them crosses the light beam, causing change of state. When it comes to wires, most of them are Just Pi cables. They are short and easy to bend and hide in enclosure. Also they are sufficiently thick to carry all currents. Standard 17AWG cables are used to connect 9V battery to the switch and power socket in Arduino itself.
- Block diagram [Fig.1. Hardware block diagram] shows how the actual device looks "on paper". Here we have all components shown, which are connected to each other exactly as on the actual device. On the second figure [Fig.2. Hardware electrical schematic] we can observe the electrical schematic with respect to all components. All the detailed information about components can be found in the bibliography.

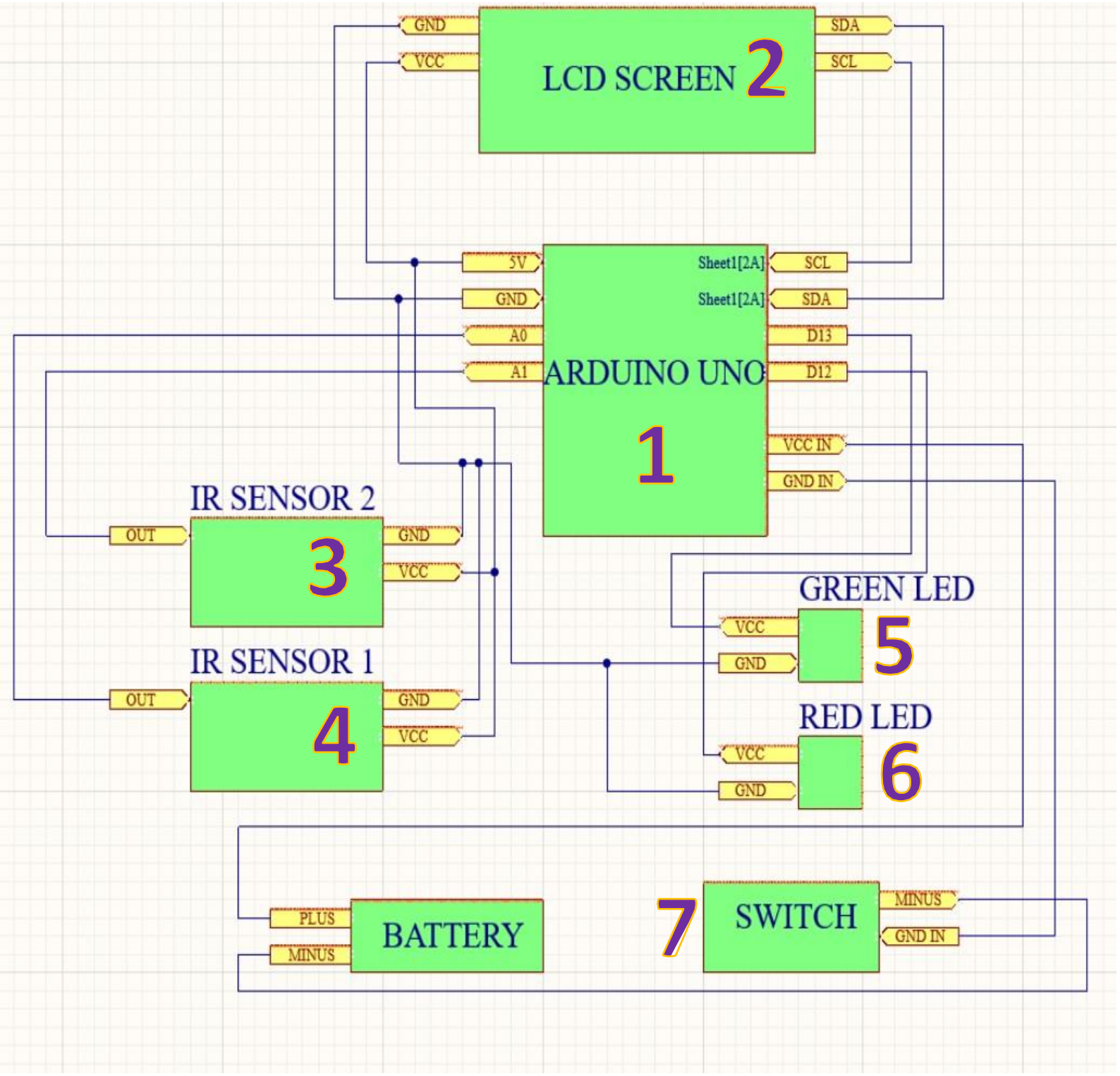


Fig.1. Hardware block diagram

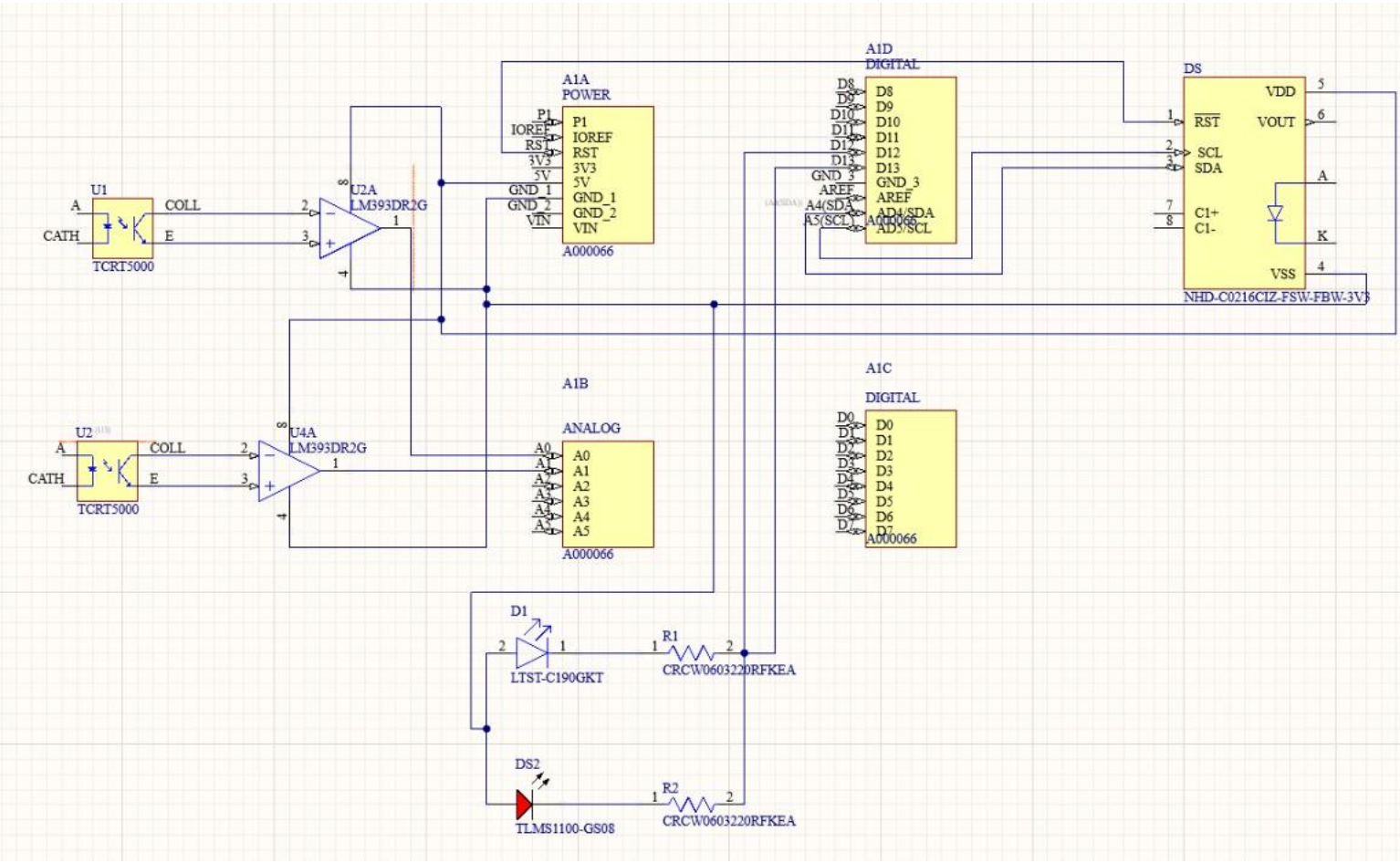


Fig.2. Hardware electrical schematic.

3.2 Actual system with reference of the block diagram:

The image bellow , show the result of the final assembly and soldering the project with all the necessary components.

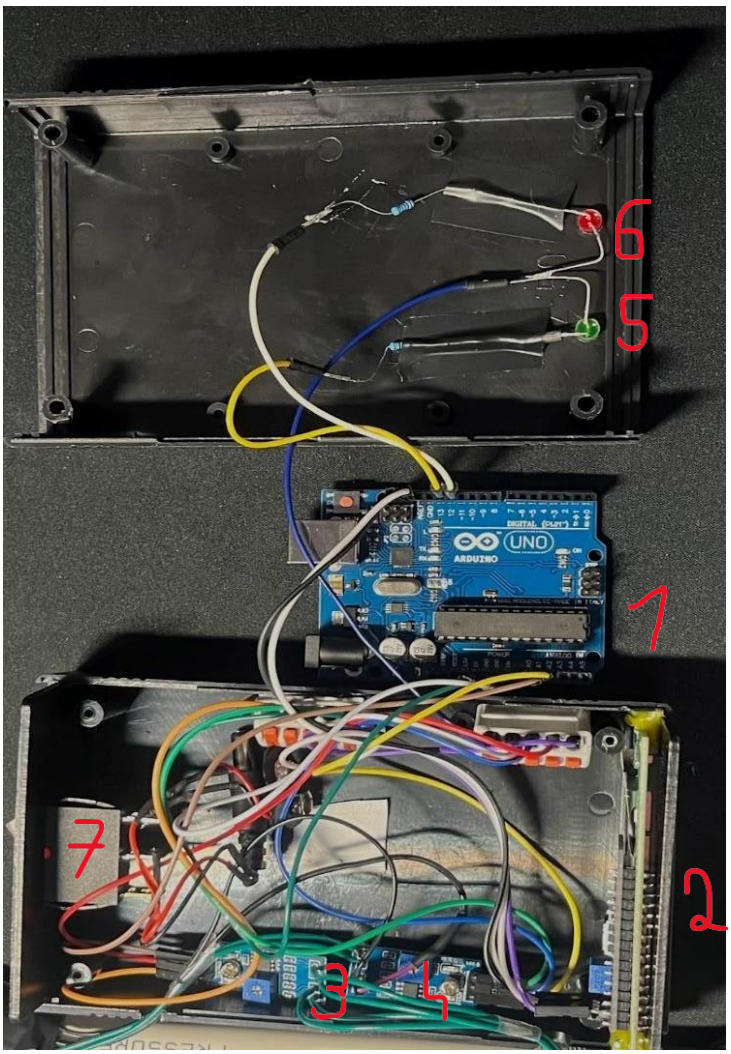


Fig.3.Real system

3.3 Key elements with references to links in bibliography:

- 2 x Optical reflective sensors module TCRT5000 LM393 IR **[1]**
- 1 x Arduino UNO **[2]**
- 1 x 9V battery **[3]**
- 1 x 9V battery socket **[4]**
- 1 x switch on-off IRS-101-1B **[5]**
- 2 x wago connectors **[6]**
- 1 x Enclosure KRADEX Z50B **[7]**
- 1 x LCD 2x16 with I2C interface **[8]**
- many JustPi cables **[9]**
- 1 x PVC tube **[10]**
- 1 x red led **[11]**
- 1 x green led **[12]**
- 2 x 220 Ohm resistors **[13]**

4. SOFTWARE DESCRIPTION:

In the following chapter we will discuss about the code. Which contains all necessary stuff for Arduino to perform calculations of the velocity, display the information on LCD screen with I2C interface and show the errors using external LEDs in the enclosure. Case of the distance was bit difficult to implement. Arduino, as its basing on Atmega, by default has inches implemented in the source. Value in the code is already converted into meters. Below there is actual code with the comments in the underbellies. Algorithm is shown below [Fig3. Program block diagram]. Actual code is shown below the algorithm. Code for this device is tested and ready to use.

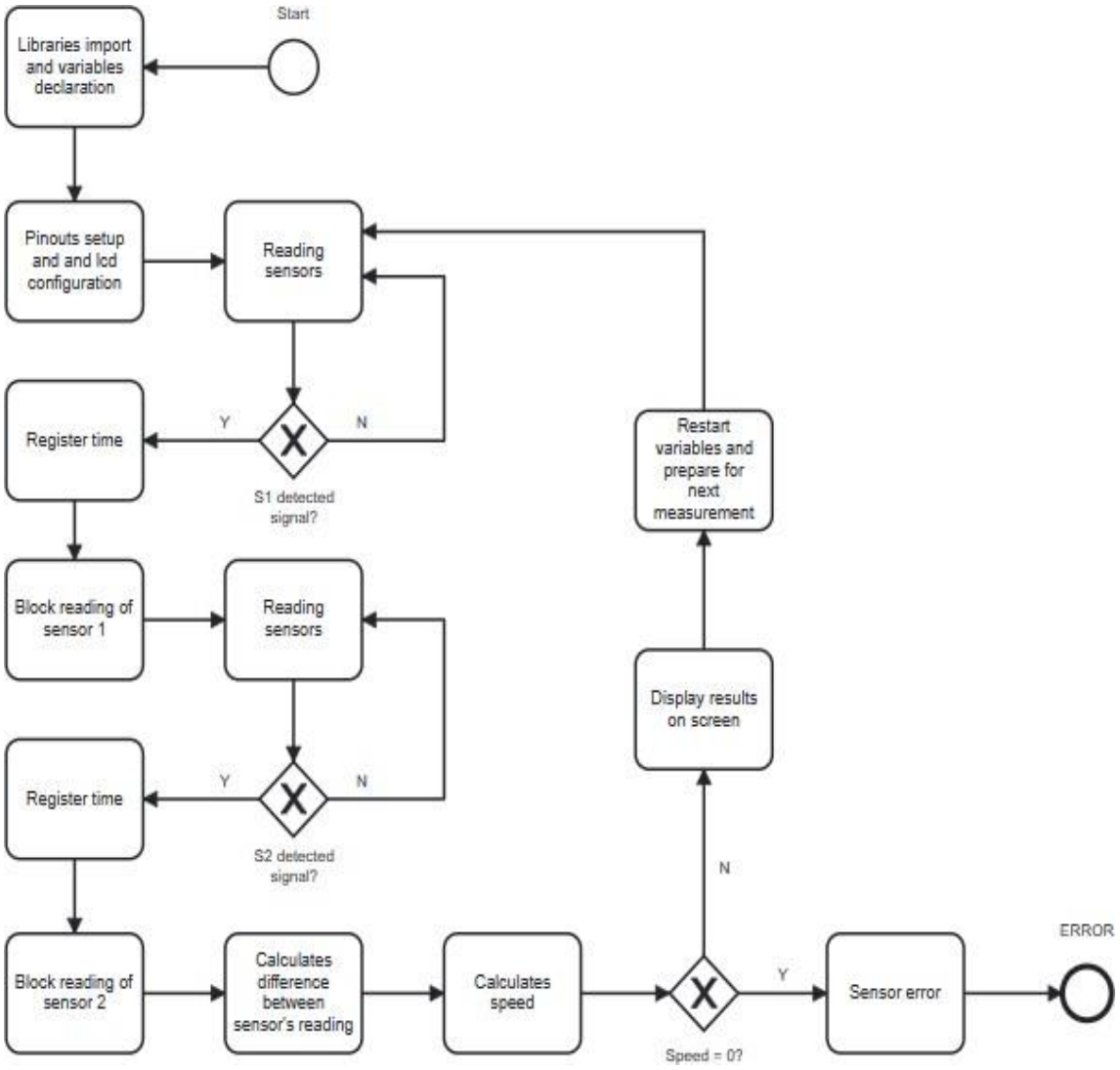


Fig.4. Fluxogram of the Algorithm

```
 _____________________________________________
|Last modified 21/01/2023                     |
|Version 20230121.v3                          |
|Code for chronometer                         |
|Author:                                      |
|Adam Stępień                                 |
|Power Supply port – 5V, ground port – GND    |
|_____________________________________________|

#include <LiquidCrystal_I2C.h> //LCD library

LiquidCrystal_I2C lcd (0x27, 16, 2); //LCD with I2C implementation

int timer1;
int timer2;
float Time;
int flag1 = 0; // reset 1
int flag2 = 0; // reset 2

float distance = 0.2030; // distance between photocells

int ir_s1 = A0; // first sensor port A0
int ir_s2 = A1; // second sensor port A1

int green_LED = 13; // green LED port 13
int red_led = 12; // red LED port 12

void setup(){
 pinMode(ir_s1, INPUT);
 pinMode(ir_s2, INPUT);
 pinMode(green_LED, OUTPUT);
 pinMode(red_led, OUTPUT);
 lcd.backlight();
 lcd.begin(16,2);
 lcd.clear();
 lcd.setCursor(0,0);
 lcd.print(" Chronograph ");
 lcd.setCursor(0,1);
 lcd.print("Optoelectronics");
 delay(2000);
lcd.clear();
}
void loop() {
if(digitalRead (ir_s1) ==  HIGH && flag1==0){timer1 = millis(); flag1=1;}
// reading first photocell
if(digitalRead (ir_s2) == HIGH && flag2==0){timer2 = millis(); flag2=1;}
//  reading second photocell
if (flag1==1 && flag2==1){
```

```
49     if(timer1 > timer2){Time = timer1 - timer2;} // comparing times between
cells and resets
50 else if(timer2 > timer1){Time = timer2 - timer1;}
51 Time=Time/1000; //convert millisecond to second
52 speed=(distance/Time); //v=d/t
53
54 }
55
56 if(speed==0){
57 lcd.setCursor(0, 1);
58 if(flag1==0 && flag2==0){lcd.print("Awaiting shot    ");}
59                     else{lcd.print("Sensor
Error    ");digitalWrite(red_led, HIGH);} // condition where
60                                         //IR_1 or IR_2 didn't read the value
61 }
62 else{
63  digitalWrite(red_led, LOW);
64    lcd.clear();
65    lcd.setCursor(0, 0);
66    lcd.print("Speed:");
67   lcd.print(speed,1);
68    lcd.print("m/s  ");
69    lcd.setCursor(0, 1);
70  if(speed > 0.5){lcd.print("   Shot counted   "); digitalWrite(green_LED,
HIGH);} // condition where both IR's read the value
71
72           else{lcd.print("                "); }
73    delay(3000);
74    digitalWrite(green_LED, LOW);
75    speed = 0;
76    flag1 = 0;
77    flag2 = 0;
78  }
79 }
```

Description :

Line 10 – including the I2C library

Lines 14-26 – declaring variables

Lines 28-41 – declaring pins and screen

Line 43–52 – main calculating function

Line 56-77 – error and not-error conditions

5. START-UP & CALIBRATION:

For the first start up and calibration, some very important steps were necessary. Firstly, calibrating the photocells. By the default, used sensors are in the configuration of proximity sensor. Both emitter and receiver diode are placed next to each other. Emitted IR beam should reach the desirable object, reflect the light beam and be received by a receiver. When this is correctly read by a sensor, the state of the device is binary 1. Here what was done, the emitter diode was unsoldered and put in front of the receiver. In this configuration, the device's state is all the time binary 1, and when the bullet crosses the light beam, it changes into binary 0. To correctly obtain the actual result, there was necessity to adjust range of the light beam using potentiometer . Secondly, correct placement of the sensors in the measurement tube. After putting them in the distance of 5 cm, the result was very unprecise. There was needed some research, which distance between the sensors will be sufficient. After combining all the factor together i.e. length of the bullet, diameter of the tube etc, best and also good looking distance between cells is circa 10 cm. Figure below [Fig 5. Start-up and calibration] shows how the setup looked like

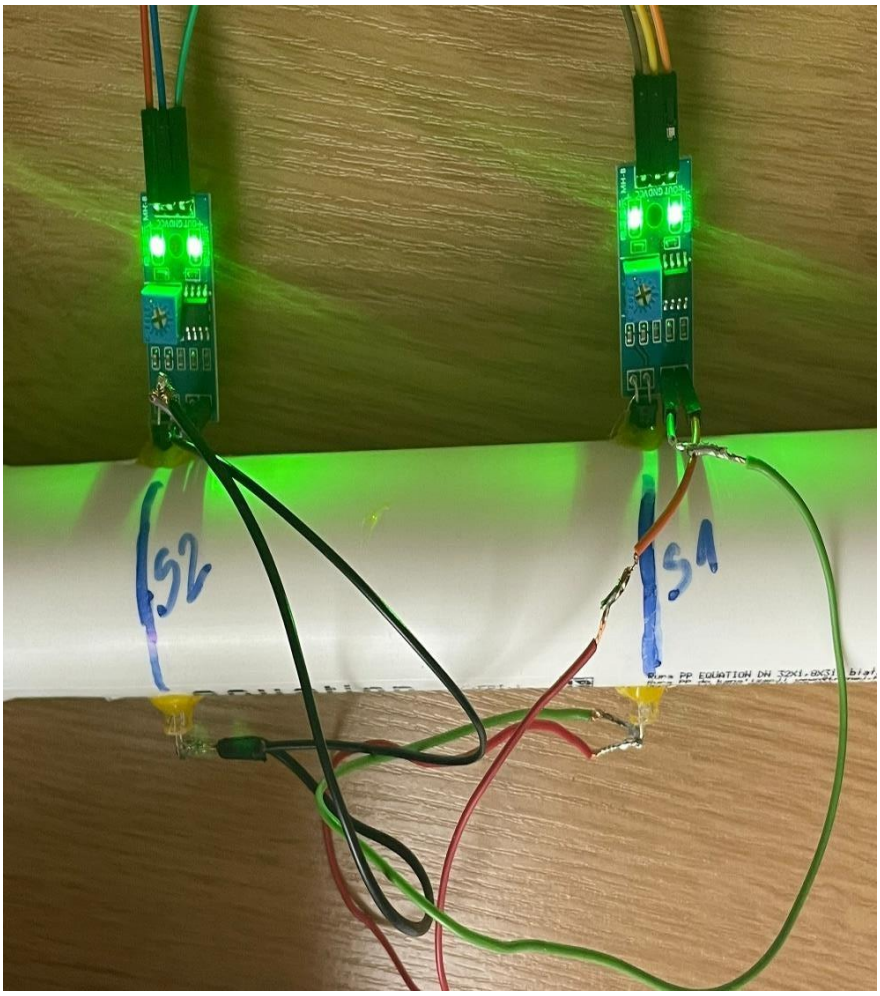


Fig.5.Star-up and calibration

6. TEST MEASUREMENTS:

For testing measurements we are using NERF foam bullet, which is sufficiently big to be read by the sensors. Its dimensions are 7 x 1 cm. Any other smaller projectile isn't wide enough to be read by used sensors. For this also, we are using the NERF gun, which nominal speed is between 30 and 50 feet per second. In conversion, this stands for 9 to 15 m/s. Used gun isn't as precise as normal firearm, but the read values are in the range. The lowest value obtained was around 10 m/s, thus the highest was 14.5 m/s. Value of 14.5 m/s is the most frequent result we obtain, so the device all in all is quite precise. All results are shown below [Table 2. Measurements results]. Details about velocity are visible in table 3 [Table 3. Calculated shot characteristics]. Table 4 shows comparison between valid and invalid measurements. Also success rate is visible there [Table 4. Device measurement information]. Graph number 1 shows measurements. More precisely blue dots are valid measurements, red are invalid and the yellow line shows average.

Shot no.	Velocity [m/s]	Shot no.	Velocity [m/s]
1	14,5	26	9,7
2	13,5	27	11,5
3	13,5	28	13,5
4	9,7	29	11,5
5	SNC	30	14,5
6	14,5	31	11,5
7	13,5	32	14,5
8	14,5	33	13,5
9	SNC	34	14,5
10	14,5	35	9,7
11	13,5	36	SNC
12	11,5	37	14,5
13	14,5	38	13,5
14	9,7	39	SNC
15	SNC	40	SNC
16	13,5	41	9,7
17	11,5	42	14,5
18	11,5	43	SNC
19	11,5	44	9,7
20	14,5	45	14,5
21	11,5	46	SNC
22	14,5	47	13,5
23	13,5	48	11,5
24	SNC	49	14,5
25	11,5	50	11,5

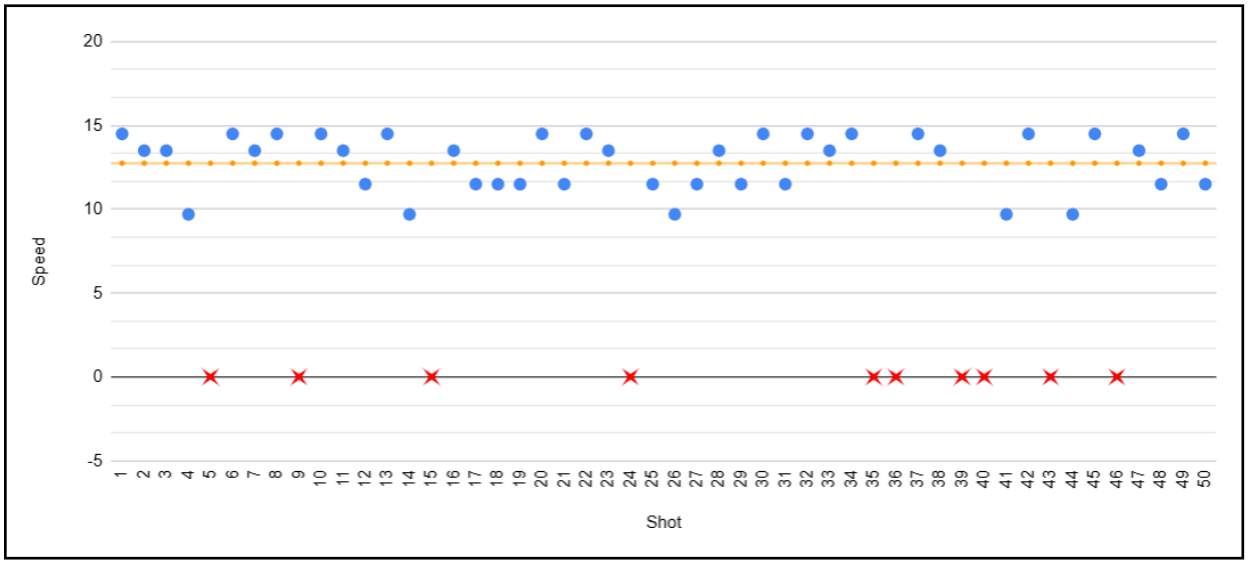
Table 2. Measurements results

Max Speed	Min Speed	Average	Standard Deviation
14,5 m/s	9,7 m/s	12,8 m/s	1,7 m/s

Table 3. Calculated shot characteristics

Valid Measurements	Invalid Measurements	Success Rate
41	9	82%

Table 4. Device measurement information



Graph 1. Measurements. Blue for valid measurements, red for invalid. Yellow line is the average

7. USER MANUAL:

1-Activate the device using on/off switch on the back.

2- On the LCD screen there is a quote, which tells when the chronograph has been activated properly, and the pop-up message will be as follow "Optoelectronics Chronograph".

3- Then a statement which says "Awaiting shot…" will show up on the screen. Meaning that the device is ready to measure.

4- Then put the outer barrel of the gun to the pipe on the left side of the device and take a shot.

5- When the shot is correctly recorded, a green LED indication it on the top of the device, which turns on for 3 seconds, and the velocity in meters per second is displayed on the LCD.

Note: while testing, something went wrong and one of the sensors could not read the bullet correctly, we have the red LED blink, and on the screen we got a message which indicates "Sensor Error". Meaning that we need to retry the shot.

8. SUMMARY:

In conclusion, we would say that this lab class and more precisely the project we had which is an optical chronograph has been an success, working on it has help us be aware of many thing, on the way of working and behaving as team.

First of all, that timing and fixing milestones is very important, and also working as a team is necessary, even though at the beginning of the project some of us did not have that much of idea about the project like how it works and what was it for, although we made some researches and discuss about it and agreed with the subject.

In the end it went absolutely good, by that, those who was confused or had different thought at some point, had the chance to ask other colleagues or the supervisor.

Each and every question was answered simply and correctly for a better understanding, we all tried our best working as a team and that of course will have an important impact on our future team projects.

Not everything was taken for granted, we had few issues all along the realisation of the project and with the chance of doing some researches before even starting the project which is very important helped a lot, we would know when and why some errors occurred.

Yet the project came up successfully done after few weeks. We had the chance to take with us new information gathered all along. Device works fine and is relatively precise. Main key for the success was unsoldering the diodes from the IR sensors, putting them next to each other and creating photocell.

9. BIBLIOGRAPHY AND LINKS TO THE ELEMENTS:

- **[1]** 2 x Optical reflective sensors module TCRT5000 LM393 IR

https://erli.pl/produkt/modul-czujnik-odbiciowy-optyczny-tcrt5000-lm393-ir-wykrywanie-przeszkod,73562857?utm_source=google&utm_medium=surface&gclid=CjwKCAiAoL6eBhA3EiwAXDom5uaf8OxjSR87qHK8HHcRJ1czQHVXVTe-3OfYGArZuKnsSJTVqslfzxoCiBsQAvD_BwE

- **[2]** 1 x Arduino UNO

https://botland.com.pl/arduino-seria-podstawowa-oryginalne-plytki/1060-arduino-uno-rev3-a000066-7630049200050.html?cd=18298825138&ad=&kd=&gclid=CjwKCAiAoL6eBhA3EiwAXDom5nd-hhRQ_dy0tGCtQt7OGBqTVGIbF5sMzDUbj3E0otMXcsE65e0KehoCPuwQAvD_BwE

- **[3]** 1 x 9V battery

https://botland.com.pl/baterie/4381-bateria-blow-super-heavy-duty-6f22-9v-5900804072973.html

- **[4]** 1 x 9V battery socket

https://botland.com.pl/zlacza-do-akumulatorow/416-klip-na-baterie-9v-6f22-z-przewodem-15cm-5903351247887.html?cd=19576772303&ad=&kd=&gclid=CjwKCAiAoL6eBhA3EiwAXDom5lc_5gT-GA_W7bXvIHa8izwi8EmqTQ-T_M-TaIpfecMq12Ntrr458hoClEoQAvD_BwE

- **[5]** 1 x switch on-off IRS-101-1B

https://eldor24.pl/adaptery-i-przejsciowki/zlacze-przelacznik-irs-101-1b-czerwony-prk0007b?gclid=CjwKCAiAoL6eBhA3EiwAXDom5gjD2UGQQqCIaNRixIFhwnhF0EM4A8-mZH394OyGA39XHQoGZqFZVhoCrs0QAvD_BwE

- **[6]** 2 x wago connectors

https://botland.com.pl/kostki-elektryczne/9991-kostka-elektryczna-5pin-250v-pomaranczowa-5904422314811.html

- **[7]** 1 x Enclosure KRADEX Z50B

https://botland.com.pl/obudowy/5927-obudowa-plastikowa-kradex-z50b-147x92x50mm-czarna-5905275013296.html?cd=18298825651&ad=&kd=&gclid=CjwKCAiAoL6eBhA3EiwAXDom5oYZR7-EZKcbX_A3NSCmwVBlGaKBSrzlBfA72GzJAgVt5u5G6IcLHxoCqQoQAvD_BwE

- **[8]** 1 x LCD 2x16 with I2C interface

https://botland.com.pl/wyswietlacze-alfanumeryczne-i-graficzne/2351-wyswietlacz-lcd-2x16-znakow-niebieski-konwerter-i2c-lcm1602-5904422309244.html?cd=18298825651&ad=&kd=&gclid=CjwKCAiAoL6eBhA3EiwAXDom5nffjyNj3khRBiKVgV3ePM65Uq2J1Ua7HLd5Q0FDxWZwvoF5DUaI2xoCHZQQAvD_BwE

- **[9]** many JustPi cables

https://botland.com.pl/przewody-polaczeniowe/19948-przewody-polaczeniowe-zensko-zenskie-justpi-10cm-40szt-5904422328672.html?cd=18298825138&ad=&kd=&gclid=CjwKCAiAoL6eBhA3EiwAXDom5glFeq2Apd5TlHoF9oNz-uwFtpkfEecGkPdKPtC77IQWQMlGhyoqahoCx1UQAvD_BwE

- **[10]** 1 x PVC tube

https://www.leroymerlin.pl/artykuly-metalowe/profile-blachy-i-akcesoria/rury-profile-okragle-i-kwadratowe/rura-okragla-pvc-1m-8x1-mm-matowa-biala-standers,p467399,l3319.html?gclid=CjwKCAiAoL6eBhA3EiwAXDom5vawJjHXg3qvt7S1BvEv65rie_W49javGHCykN1GIbcha20xo57cpRoCWoAQAvD_BwE&gclsrc=aw.ds

- **[11]** 1 x red led

https://botland.com.pl/diody-led/19995-dioda-led-5mm-czerwona-10szt-justpi-5904422328832.html

- **[12]** 1 x green led

led https://botland.com.pl/diody-led/19987-dioda-led-3mm-zielona-10szt-justpi-5904422328757.html

- **[13]** 2 x 220 Ohm resistors

https://www.mouser.pl/ProductDetail/YAGEO/CFR-25JB-52-470R?qs=oypCK0zG326lJod0wvnCOg%3D%3D&mgh=1&vip=1&gclid=CjwKCAiAoL6eBhA3EiwAXDom5ruUFZNVucOXsvLJ50S17rDN2OiF0DQTfftFQmERxzRBIeE_MATqYhoCgEIQAvD_BwE

Reference bibliography links

https://www.analog.com/

https://www.arduino.cc/

https://github.com/

https://pl.wikipedia.org/

https://botland.com.pl