Door unlocker interphone type

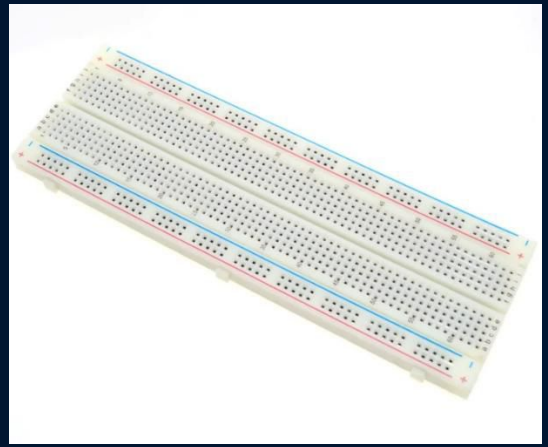
MeasurEments and actuators

Presented by Bejenariu Adina
Lupu Dima Daria
Pleșca Evelyn

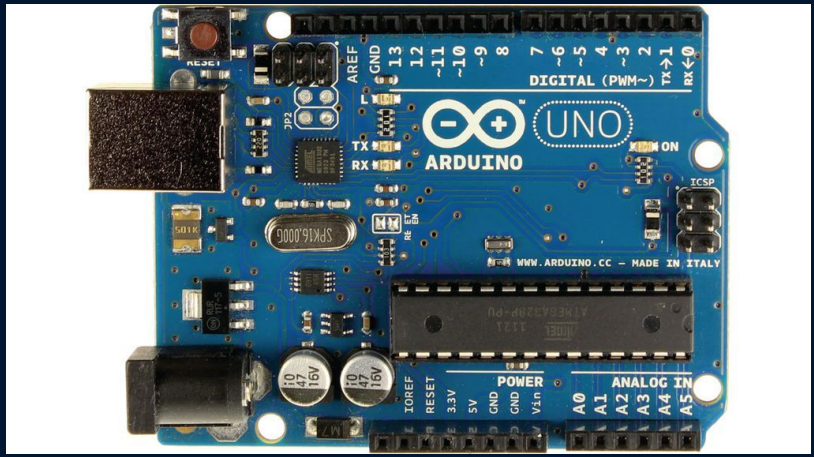
What we used

The main idea is the design of a mechanism that simulates the process of unlocking an intercom.

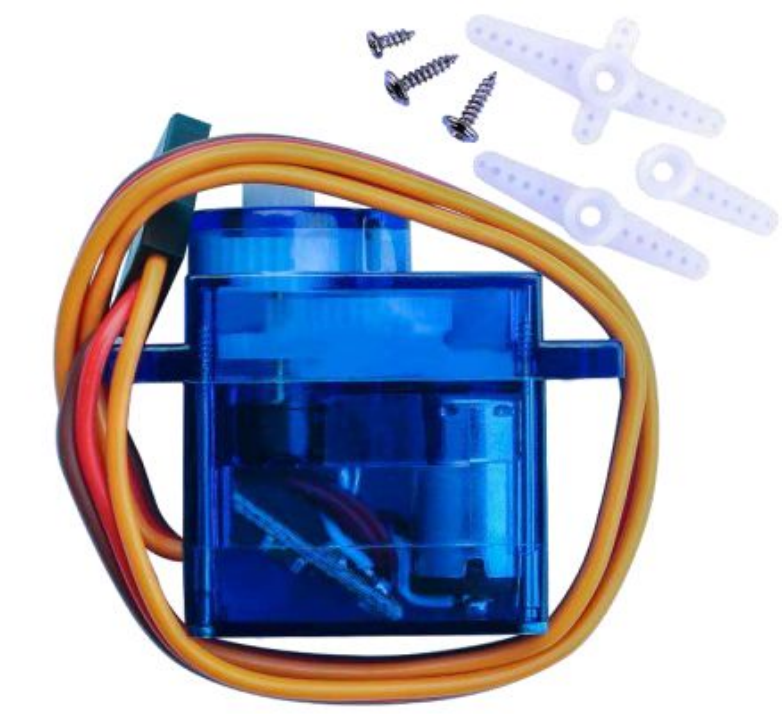
For the design mechanism the next components were used:



Breadboard to create the circuits



Motherboard Arduino UNO

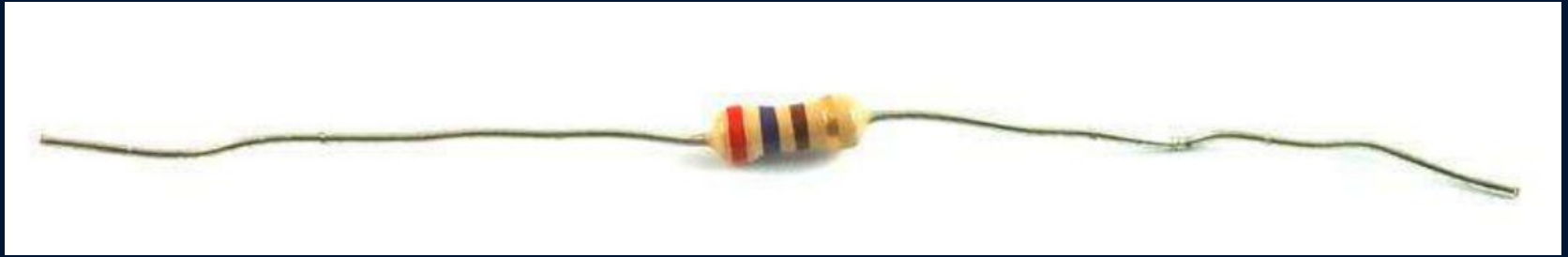


Servo motor SG90 used to simulate the door opening

LCD I2C with 4 pins and integrated potentiometer

Rc522 RFID Module for personalized scanning

A passive buzzer used for sound



2 resistors of 220 ohm

Ultrasonic sensor HC-SR04

Used LED s

Male-to-male wires and
female-to-male wires

HOW DOES IT WORK?

After the programme is activated the LCD display doesn’t show anything.

When the ultrasonic sensor senses an object close to it (distance 30cm or less) the LCD display shows the message: Buna ziua Prezenta□i cardu(l).

If the ultrasonic sensor senses an object close to it and the electromagnetic card is put next to the RFID module, the SG90 servo motor opens the door and the green LED turns on and the buzzer makes a specific sound.

If the ultrasonic sensor doesn’t sense an object close to it and the electromagnetic card is put next to the RFID module, the SG90 servo motor doesn’t open the door and the red LED stays on.

```
#include <LiquidCrystal_I2C.h>
#include <Servo.h>
#include <SPI.h>
#include <MFRC522.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

#define SS_PIN 10
#define RST_PIN 9

#define redLed_pin 4
#define greenLed_pin 2
#define echoPin 7
#define trigPin 5

const int enterCode[] = { 113, 124, 95, 8 };
long duration;
int distance;

LiquidCrystal_I2C lcd(0x3F, 16, 2);
MFRC522 rfid(SS_PIN, RST_PIN); // Instance of the class

MFRC522::MIFARE_Key key;

// Init array that will store new NUID
byte nuidPICC[4];

Servo lacatUsa;

void setup() {

    Serial.begin(9600);
    SPI.begin(); // Init SPI bus
    rfid.PCD_Init(); // Init MFRC522

    for (byte i = 0; i < 6; i++) {
        key.keyByte[i] = 0xFF;
    }
    pinMode(6, OUTPUT);
    pinMode(redLed_pin, OUTPUT);
    pinMode(greenLed_pin, OUTPUT);
    digitalWrite(redLed_pin, HIGH);

    lacatUsa.attach(3);

    Serial.println("Initializare...");
    lacatUsa.write(50);

    pinMode(trigPin, OUTPUT); // Sets the trigPin as an OUTPUT
    pinMode(echoPin, INPUT); // Sets the echoPin as an INPUT
    Serial.begin(9600); // // Serial Communication is starting with 9600 of baudrate speed

    lcd.init();
    lcd.backlight();
    lcd.clear();----------------------------------------------------
    lcd.setCursor(4,0);
    lcd.print("Hackster");

}
```

```
void loop() {

    // Clears the trigPin condition
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    // Sets the trigPin HIGH (ACTIVE) for 10 microseconds
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    // Reads the echoPin, returns the sound wave travel time in microseconds
    duration = pulseIn(echoPin, HIGH);
    // Calculating the distance
    distance = duration * 0.034 / 2; // Speed of sound wave divided by 2 (go and back)
    // Displays the distance on the Serial Monitor
    Serial.print("Distance: ");
    Serial.print(distance);
    Serial.println(" cm");

    if (distance <=30){
          lcd.clear();
          lcd.print("Bine ati venit!");
          lcd.setCursor(0,1);
          lcd.print("Prezentati cardul va rog!");

        if (!rfid.PICC_IsNewCardPresent()) {
            return;
        }


        if (!rfid.PICC_ReadCardSerial())
            return;


        MFRC522::PICC_Type piccType = rfid.PICC_GetType(rfid.uid.sak);
```

```
// Check is the PICC of Classic MIFARE type
if (piccType != MFRC522::PICC_TYPE_MIFARE_MINI &&
    piccType != MFRC522::PICC_TYPE_MIFARE_1K &&
    piccType != MFRC522::PICC_TYPE_MIFARE_4K) {
    Serial.println(F("Your tag is not of type MIFARE Classic."));
    return;
}


Serial.println(F("Card detectat"));
int enter = 0;

for (byte i = 0; i < 4; i++) {
    nuidPICC[i] = rfid.uid.uidByte[i];

    if (nuidPICC[i] == enterCode[i]) {
        enter++;
        Serial.println(nuidPICC[i]);
    }
}
if (enter == 4) {

    Serial.println("Acces permis");
    lcd.clear();
    lcd.print("Acces permis!");
    lcd.setCursor(0,1);
    lcd.print("Va uram o zi buna!");
    openDoor();
    delay(400);
}
else {
    lcd.clear();
    lcd.print("Acces respins!");
    lcd.setCursor(0,1);
    lcd.print("Hai pa!");
    Serial.println("Acces nepermis");
    delay(1000);
}
Serial.println();
//Serial.println("Apropiati cardul pentru citire");
```

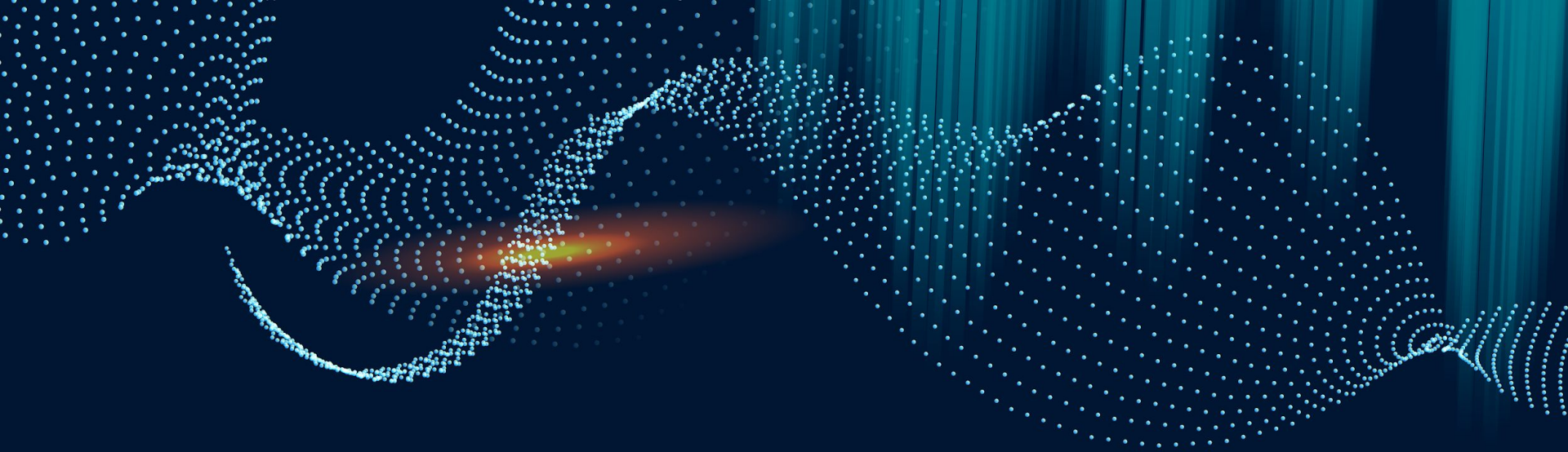
```
        // Halt PICC
        rfid.PICC_HaltA();

        // Stop encryption on PCD
        rfid.PCD_StopCryptol();
        }
        if (distance >30)
        lcd.clear();
}


/**
 * Helper routine to dump a byte array as hex values to Serial.
 */
void printHex(byte* buffer, byte bufferSize) {
    for (byte i = 0; i < bufferSize; i++) {
        Serial.print(buffer[i] < 0x10 ? " 0" : " ");
        Serial.print(buffer[i], HEX);
    }
}

/**
 * Helper routine to dump a byte array as dec values to Serial.
 */
void printDec(byte* buffer, byte bufferSize) {
    for (byte i = 0; i < bufferSize; i++) {
        Serial.print(buffer[i] < 0x10 ? " 0" : " ");
        Serial.print(buffer[i], DEC);
    }
}
```

```
void openDoor() {
    digitalWrite(redLed_pin, LOW);
    digitalWrite(greenLed_pin, HIGH);
    lacatUsa.write(100);
    tone(6, 1000);
    delay(2000);
    noTone(6);
    digitalWrite(redLed_pin, HIGH);
    digitalWrite(greenLed_pin, LOW);
    lacatUsa.write(10);
}
```



Components Explained

Description of the sensors and the actuators used

Ultrasonic Sensor

The main purpose of the ultrasonic (proximity) sensors is to detect the presence of objects without physical contact. An ultrasonic sensor emits an electromagnetic field or infrared rays, then it waits for changes in the electromagnetic field or it waits for a return signal.

Ultrasonic sensor module HC-SR04 provides 2cm-400cm non-contact measurement function, the ranging accuracy can reach to 3mm. The modules includes ultrasonic transmitters, receiver and control circuit. The basic principle of work:

1 - The Trigger Pin emits a signal
2 - The Echo pin waits for the signal to come back
3 - The sensor counts how much time it takes for the signal it emitted to go back, and based on that number it we use the following formula to find the distance

```
distance = duration * 0.034 / 2;
```

<- the wave travels back and forth

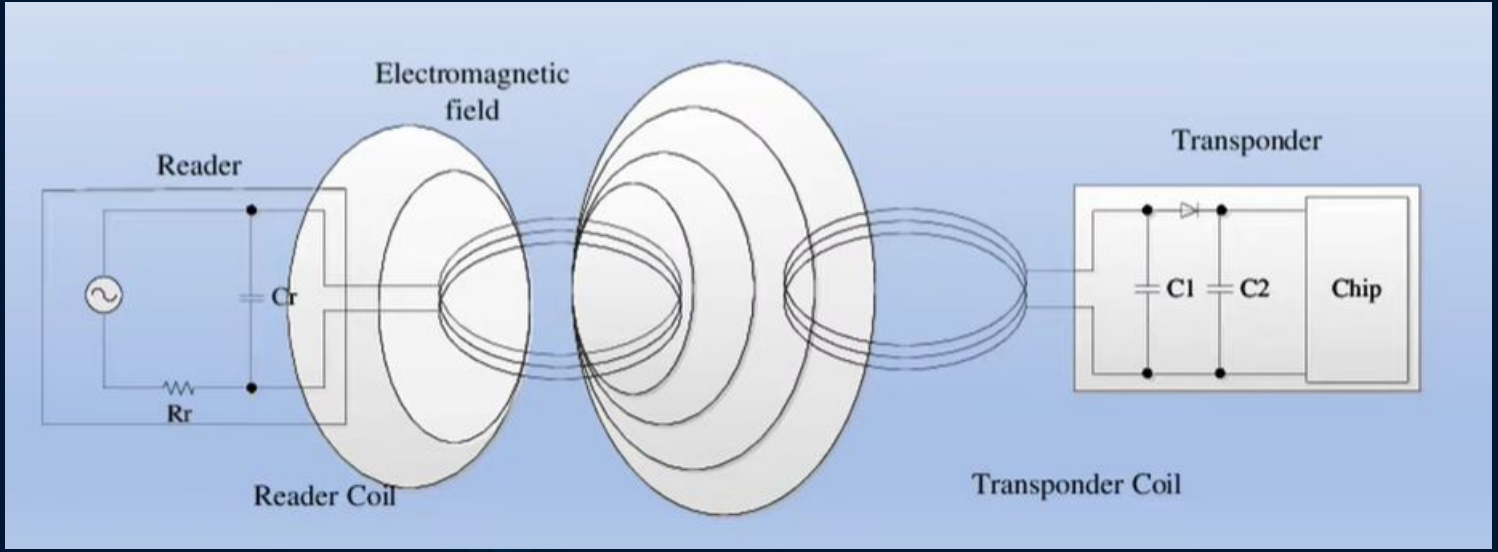
RFID Module

Radio-frequency identification (RFID) uses electromagnetic fields to automatically identify and track tags attached to objects. An RFID system consists of a tiny radio transponder, a radio receiver and transmitter. When triggered by an electromagnetic interrogation pulse from a nearby RFID reader device, the tag transmits digital data back to the reader.

In simple words an RFID uses electromagnetic fields to transfer data over short distances. RFID is useful to identify people, to make transactions, etc…

RFID module - MFRC522

How a RFID module works



Servo motor SG90

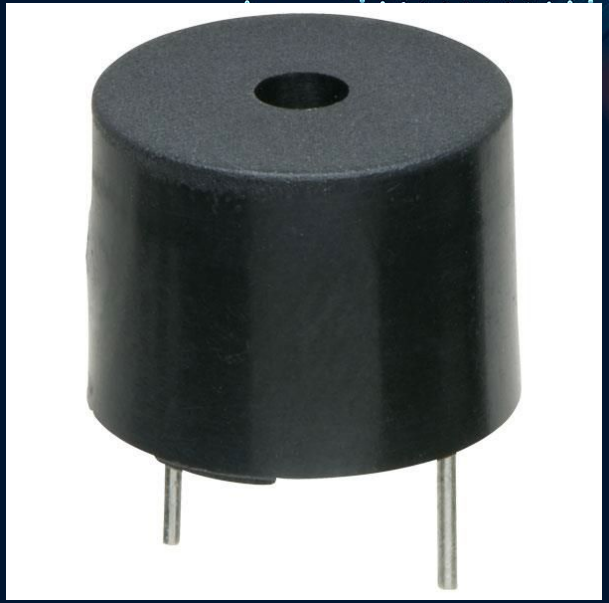
Servo is a type of geared motor that can only rotate 180 degrees. It is controlled by sending electrical pulses from your UNO board. These pulses tell the servo what position it should move to. The Servo has three wires, of which the brown one is the ground wire and should be connected to the GND port of UNO, the red one is the power wire and should be connected to the 5v port, and the orange one is the signal wire and should be connected to the Dig #9 port.

The servo motor SG90 used for the door unlocker

Passive Buzzer

The working principle of a passive buzzer is to use a PWM signal(Pulse Width Modulation) that generates sound to create vibrations in the air.

Modified according to the length of the vibration frequency, it can generate different sounds. For example, for a 523 Hz pulse, it can generate the Alto Do note, for a 587 Hz pulse, it can generate the Re note. With the help of the buzzer you can create a song.

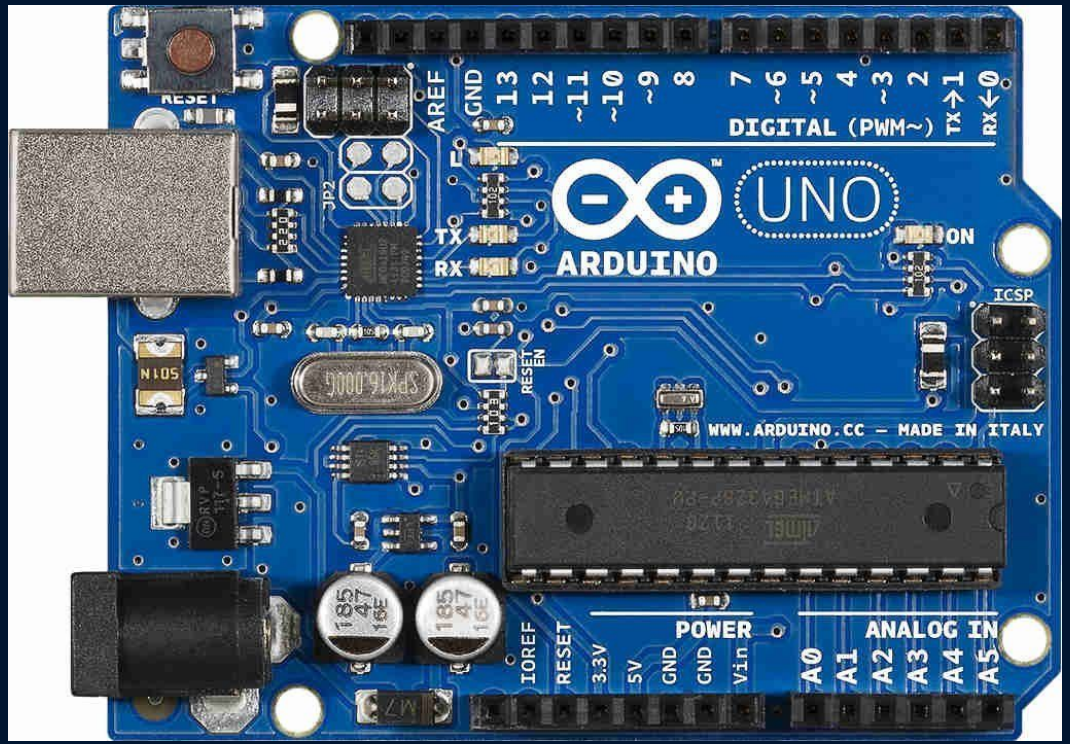


A passive buzzer used for sound

LCD I2C

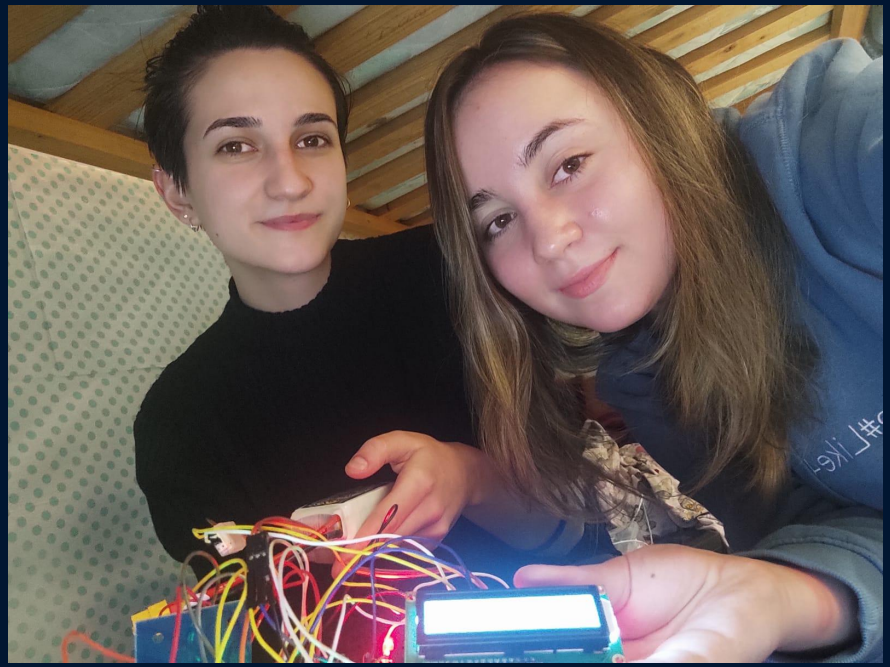
The LCD IC2 is a LCD with the potentiometer already connected to the display. In consequence, it has only 4 pins to connect with the motherboard.

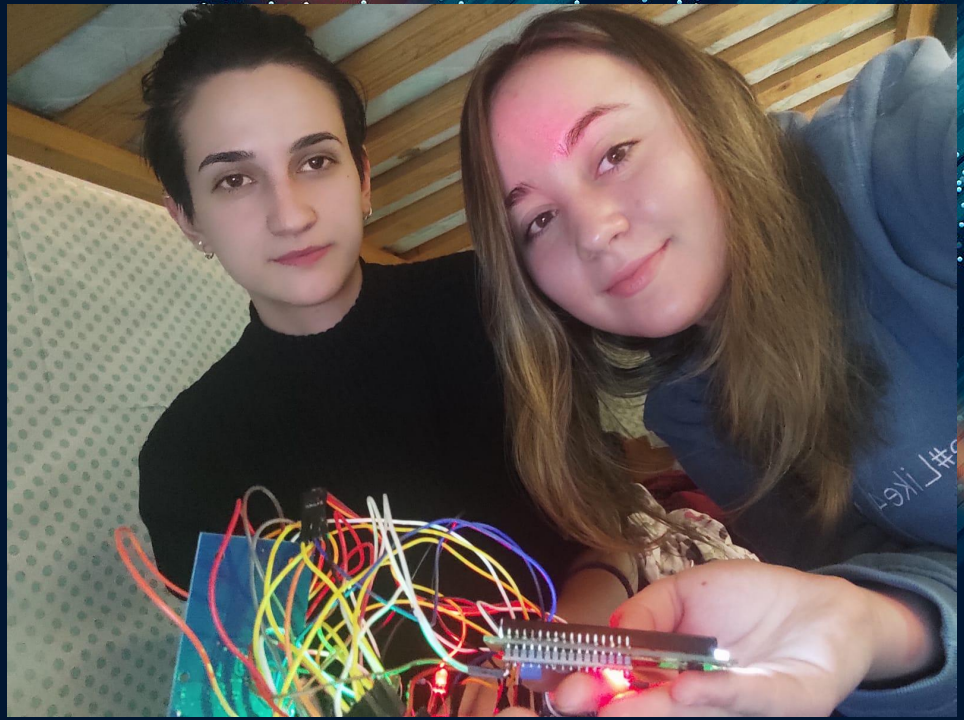
Liquid Crystal Display I2C


RESET
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM~)
UNO
ARDUINO
ON
ICSP
TX
RX
RESET
WWW.ARDUINO.CC – MADE IN ITALY
POWER
ANALOG IN
IOREF
RESET
3.3V
5V
GND
GND
Vin
A0
A1
A2
A3
A4
A5

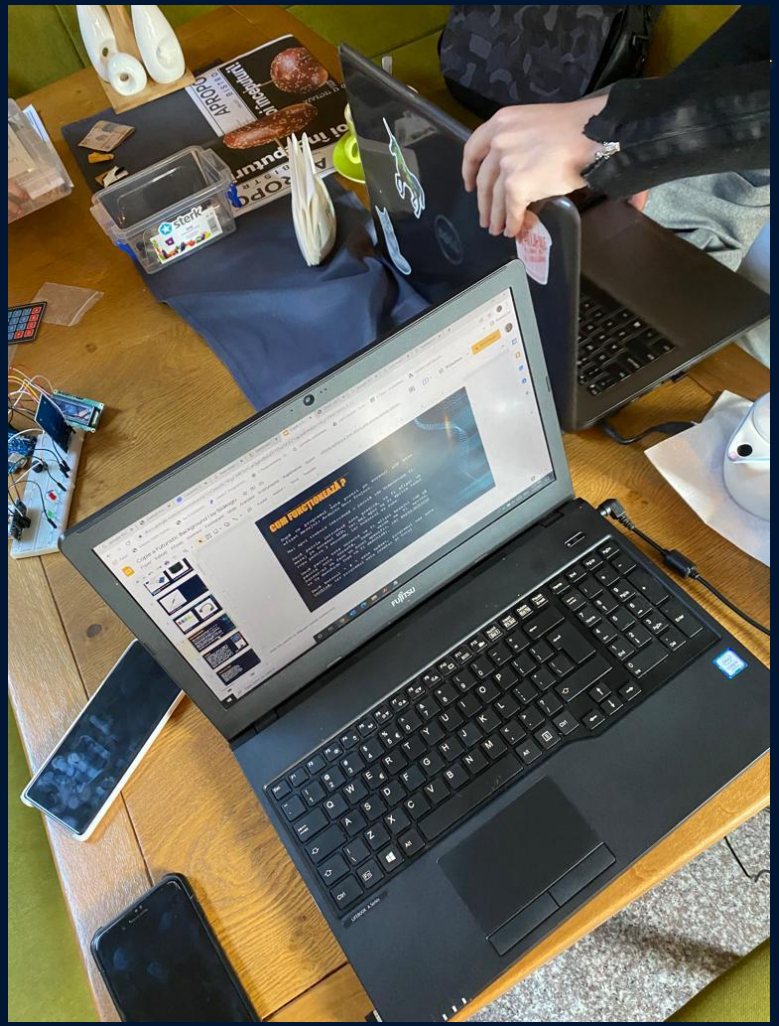
Ardulno UNO

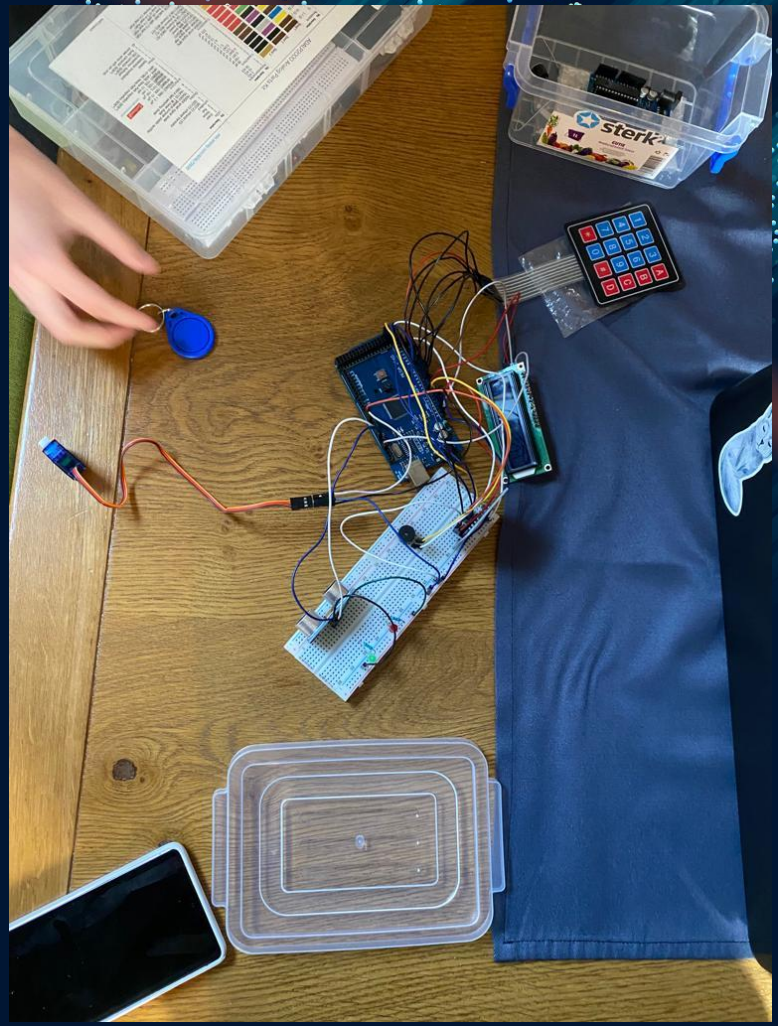
The process of making:






MANCARE


sterk
Fujitsu

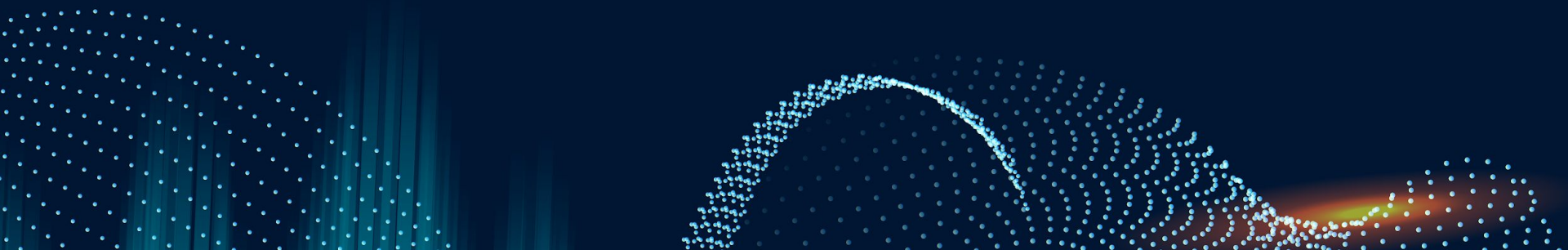

sterk

A video demonstration here :

https://drive.google.com/file/d/1cHPONS5QSmSHxG1cl3XyhdReoxNWwDpS/view?usp=sharing

Our code can be found here:

https://github.com/plescaevelyn/door-unlocker-interphone



BIBLIOGRAFIE

- https://images-na.ssl-images-amazon.com/images/I/D1oC-c3G5TS.pdf
- https://www.youtube.com/watch?v=FwbWvjq_iiM