Práctica 5: Arduino-Elegoo

Estructura de Computadores 2021-2022

Ejercicio 1: Blink

Código arduino:

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);  // turn the LED on (HIGH is the voltage
level)
  delay(100);                       // wait for a second  MODIFICAD0: DE 1000 A
100
  digitalWrite(LED_BUILTIN, LOW);   // turn the LED off by making the voltage LOW
  delay(100);                       // wait for a second  MODIFICADO: DE 1000 A
100
}
```

Código ensamblador:

Se han modificado los tiempos: el led se enciende por 100ms y queda apagado durante 1000ms

```
void my_delay(uint16_t ms);

#define CPU_FREQUENCY 16000000UL
uint16_t delay_count = CPU_FREQUENCY / 4000;

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  // pinMode(LED_BUILTIN, OUTPUT);
  asm volatile (
    "sbi %0, %1 \n"
    : : "I" (_SFR_IO_ADDR(DDRB)), "I" (DDB7)
  );
}

// the loop function runs over and over again forever
void loop() {
  // digitalWrite(LED_BUILTIN, HIGH);// turn the LED on (HIGH is the voltage
level)
  // digitalWrite(13          , HIGH);// turn the LED on (HIGH is the voltage
level)

  asm volatile (
    "sbi %0, %1 \n"
```

```
    : : "I" (_SFR_IO_ADDR(PORTB)),"I" (PORTB7)
  );

  my_delay(100);   // wait for 10 ms   MODIFICADO A 100

  // digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage
LOW
  // digitalWrite(13         , LOW); // turn the LED off by making the voltage
LOW

  asm volatile (
    "cbi %0, %1 \n"
    : : "I" (_SFR_IO_ADDR(PORTB)),"I" (PORTB7)
  );

  my_delay(1000);  // wait for 990 ms  MODIFICADO A 1000
}

void my_delay(uint16_t ms) {
  uint16_t cnt;

  asm volatile (
                      "\n"
    "loop_ms:"        "\n\t"
      "mov %A0, %A2"  "\n\t"
      "mov %B0, %B2"  "\n"

    "loop_cnt:"       "\n\t"
      "sbiw %A0, 1"   "\n\t"
      "brne loop_cnt" "\n\t"

      "sbiw %1,  1"   "\n\t"
      "brne loop_ms"  "\n\t"

    : "=&w" (cnt)
    : "w" (ms), "r" (delay_count)
  );
}
```

Esquemático:

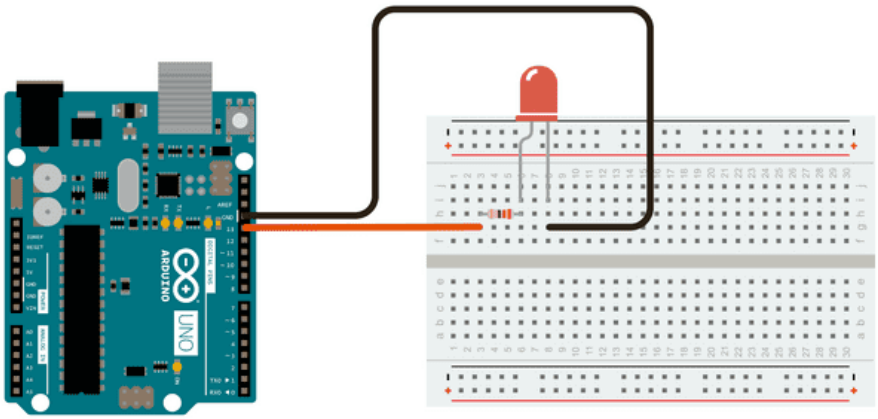
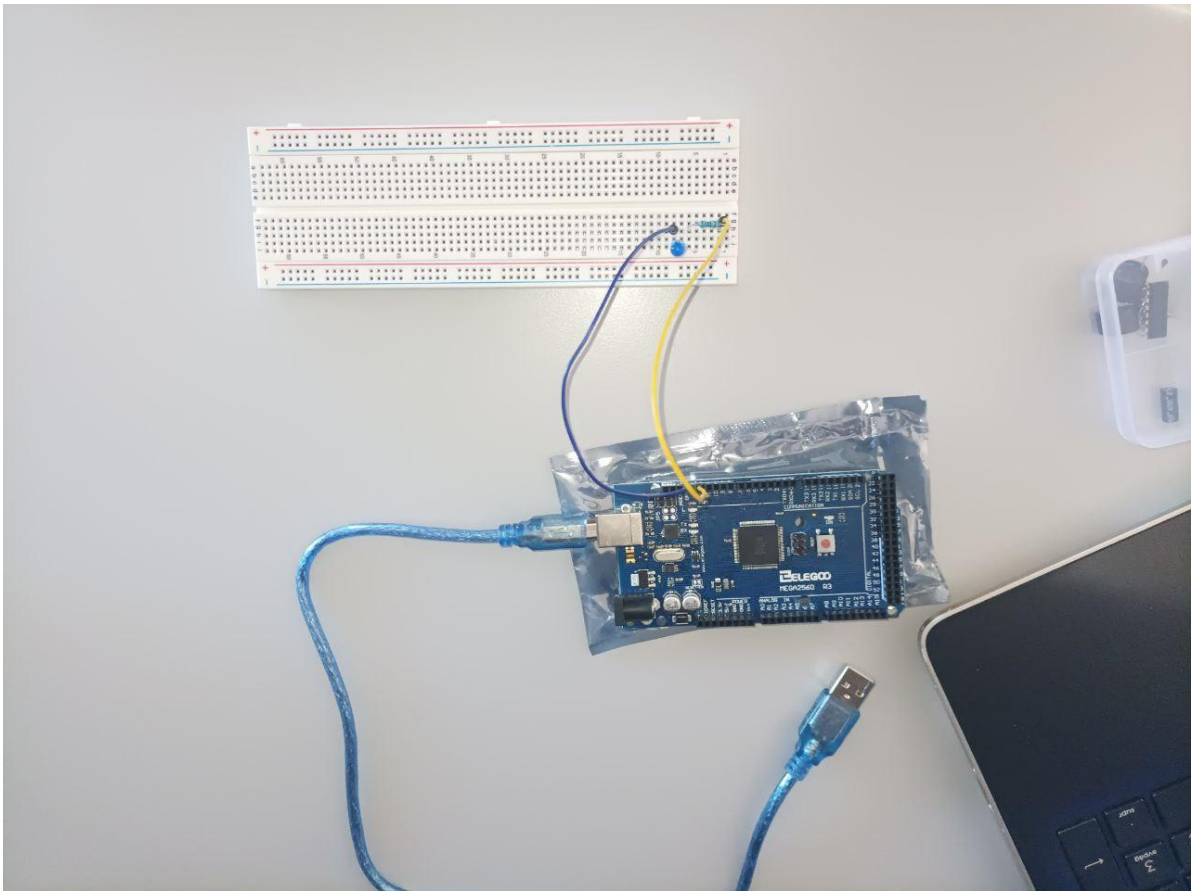


Foto:



Vídeo:

Compartido en Drive (acceso limitado a cuentas go.ugr):

https://drive.google.com/file/d/1m7X7uZxuHPpD15nqbpexG8Iy3PCr6PFQ/view?usp=sharing

Ejercicio 2: Zumbador pasivo

Código arduino:

Código y melodía tomados de robsoncouto@Github:

https://github.com/robsoncouto/arduino-songs/blob/master/professorlayton/professorlayton.ino

Esquemático:

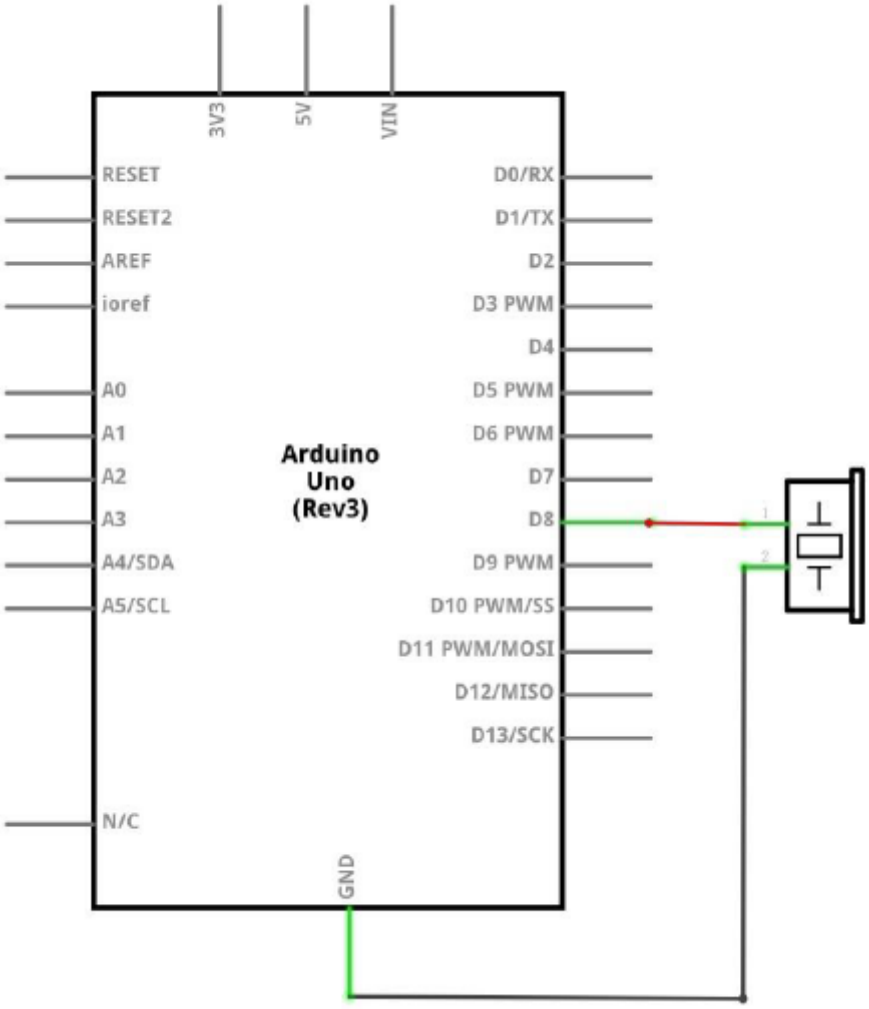
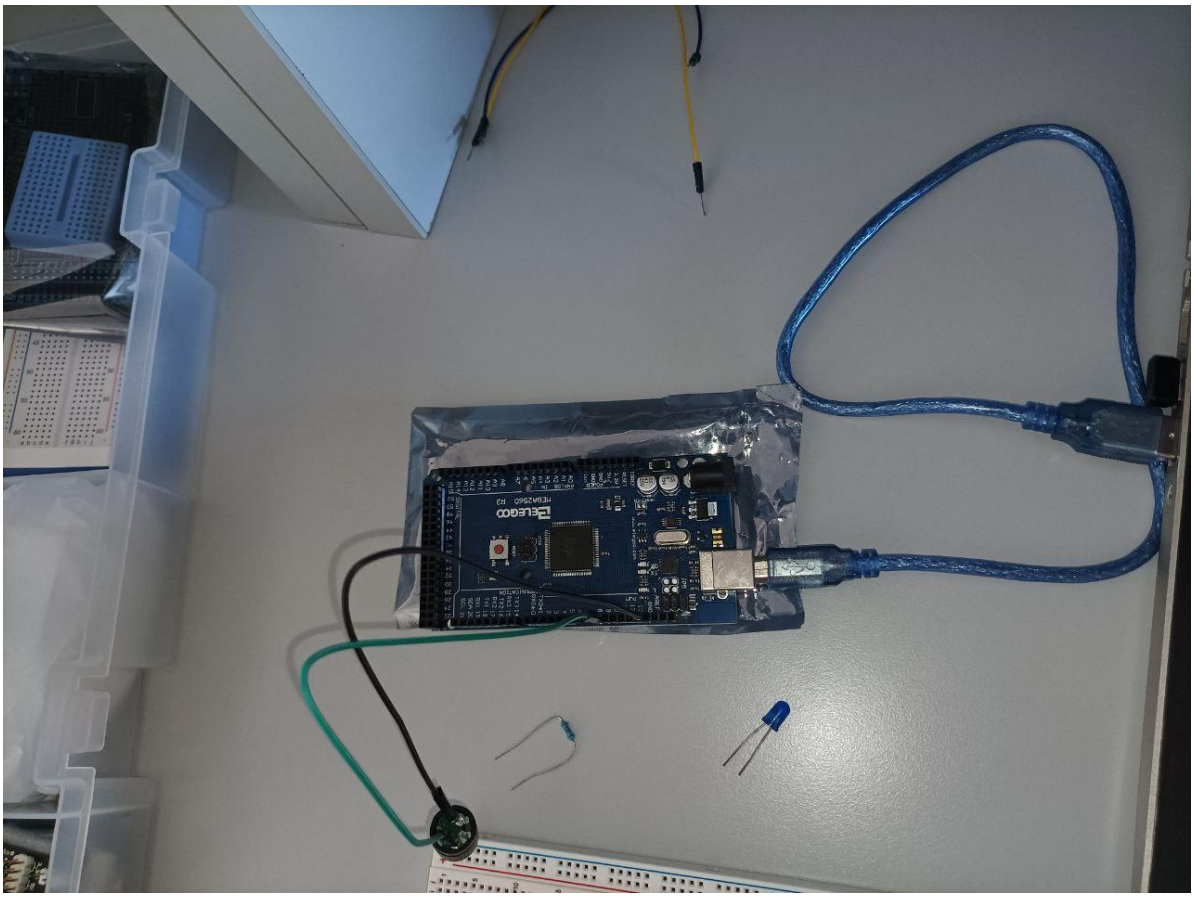


Foto:



Vídeo:

Compartido en Drive (acceso limitado a cuentas go.ugr):

https://drive.google.com/file/d/1DahwgH7xuidZb0NUJnEM_Fe6D6idsrts/view?usp=sharing

Ejercicio 3: Theremin de leds

Código Arduino:

```
//www.elegoo.com
//2016.12.9

int lightPin = 0;
int latchPin = 11;
int clockPin = 9;
int dataPin = 12;

int leds = 0;

void setup()
{
  pinMode(latchPin, OUTPUT);
  pinMode(dataPin, OUTPUT);
  pinMode(clockPin, OUTPUT);
}
void updateShiftRegister()
{
   digitalWrite(latchPin, LOW);
   shiftOut(dataPin, clockPin, LSBFIRST, leds);
   digitalWrite(latchPin, HIGH);
}
void loop()
{
  int reading  = analogRead(lightPin);
  int numLEDSLit = reading / 57;  //1023 / 9 / 2
  if (numLEDSLit > 8) numLEDSLit = 8;
  leds = 0;   // no LEDs lit to start
  for (int i = 0; i < numLEDSLit; i++)
  {
    leds = leds + (1 << i);  // sets the i'th bit
  }
  updateShiftRegister();
}
```

Esquemático:

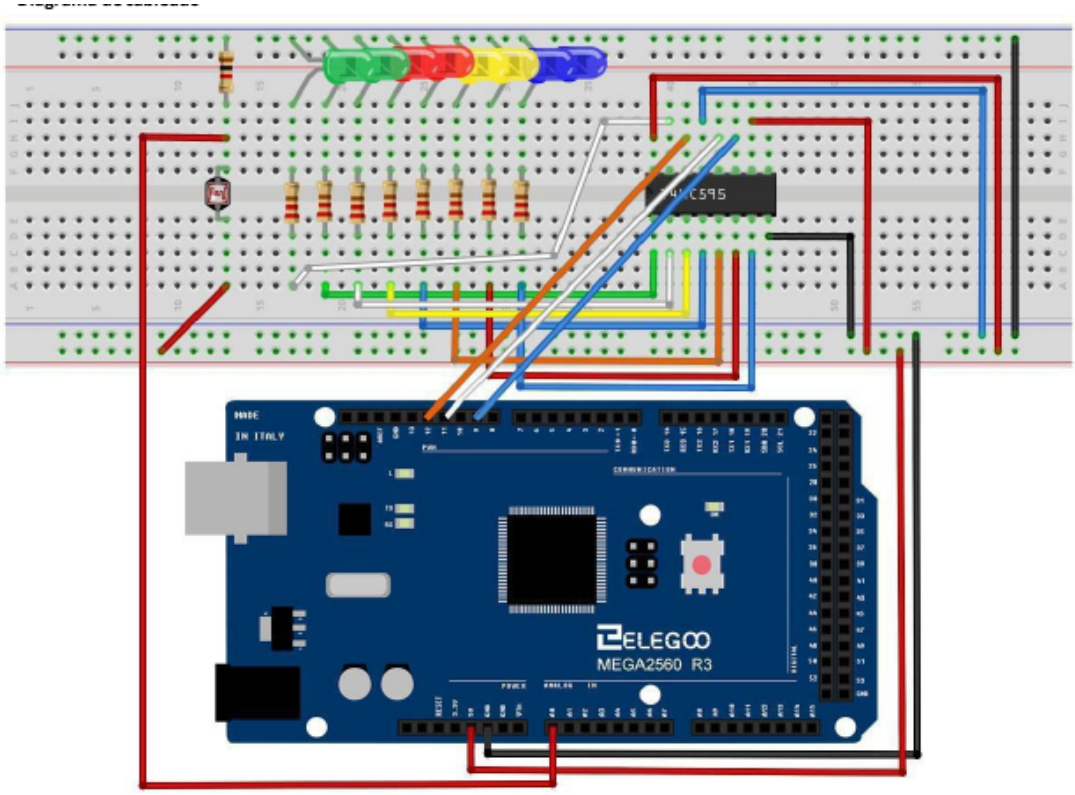
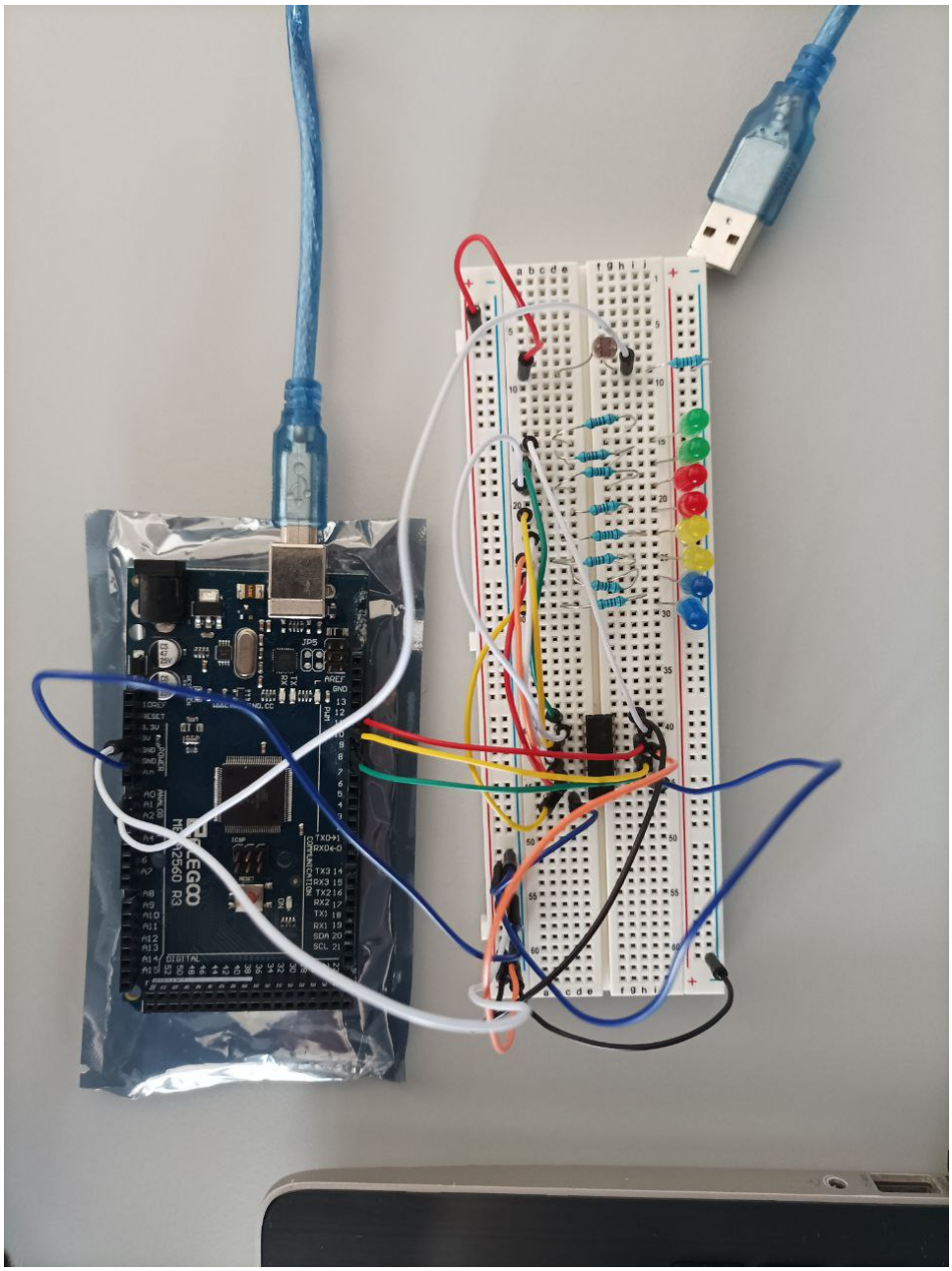


Foto:



Vídeo:

Compartido en Drive (acceso limitado a cuentas go.ugr):

https://drive.google.com/file/d/1iqAHDm0gi15dGfS8Y_r2EeqEh7pQBkxk/view?usp=sharing

Ejercicio 4: Theremin de leds y zumbador

Código Arduino:

```
int sensorValue; //int lightPin = 0;
int latchPin = 11;
int clockPin = 9;
int dataPin = 12;

// variable to calibrate low value
int sensorLow = 1023;
// variable to calibrate high value
int sensorHigh = 0;
// LED pin
const int ledPin = 13;

int leds = 0;

void setup()
{
  pinMode(latchPin, OUTPUT);
  pinMode(dataPin, OUTPUT);
  pinMode(clockPin, OUTPUT);

  // Make the LED pin an output and turn it on
  pinMode(ledPin, OUTPUT);
  digitalWrite(ledPin, HIGH);

  // calibrate for the first five seconds after program runs
  while (millis() < 5000) {
    // record the maximum sensor value
    sensorValue = analogRead(A0);
    if (sensorValue > sensorHigh) {
      sensorHigh = sensorValue;
    }
    // record the minimum sensor value
    if (sensorValue < sensorLow) {
      sensorLow = sensorValue;
    }
  }
  // turn the LED off, signaling the end of the calibration period
  digitalWrite(ledPin, LOW);
}
void updateShiftRegister()
{
   digitalWrite(latchPin, LOW);
   shiftOut(dataPin, clockPin, LSBFIRST, leds);
   digitalWrite(latchPin, HIGH);
}



void loop()
{
  int reading  = analogRead(sensorValue);
  int numLEDSLit = reading / 120;  //1023 / 9 / 2
  if (numLEDSLit > 8) numLEDSLit = 8;
```

```
  leds = 0;    // no LEDs lit to start
  for (int i = 0; i < numLEDSLit; i++)
  {
    leds = leds + (1 << i);  // sets the i'th bit
  }
  updateShiftRegister();

  //read the input from A0 and store it in a variable
  sensorValue = analogRead(A0);

  // map the sensor values to a wide range of pitches
  int pitch = map(sensorValue, sensorLow, sensorHigh, 50, 4000);

  // play the tone for 20 ms on pin 8
  tone(8, pitch, 10);

  // wait for a moment
  delay(10);
}
```

Esquemático:

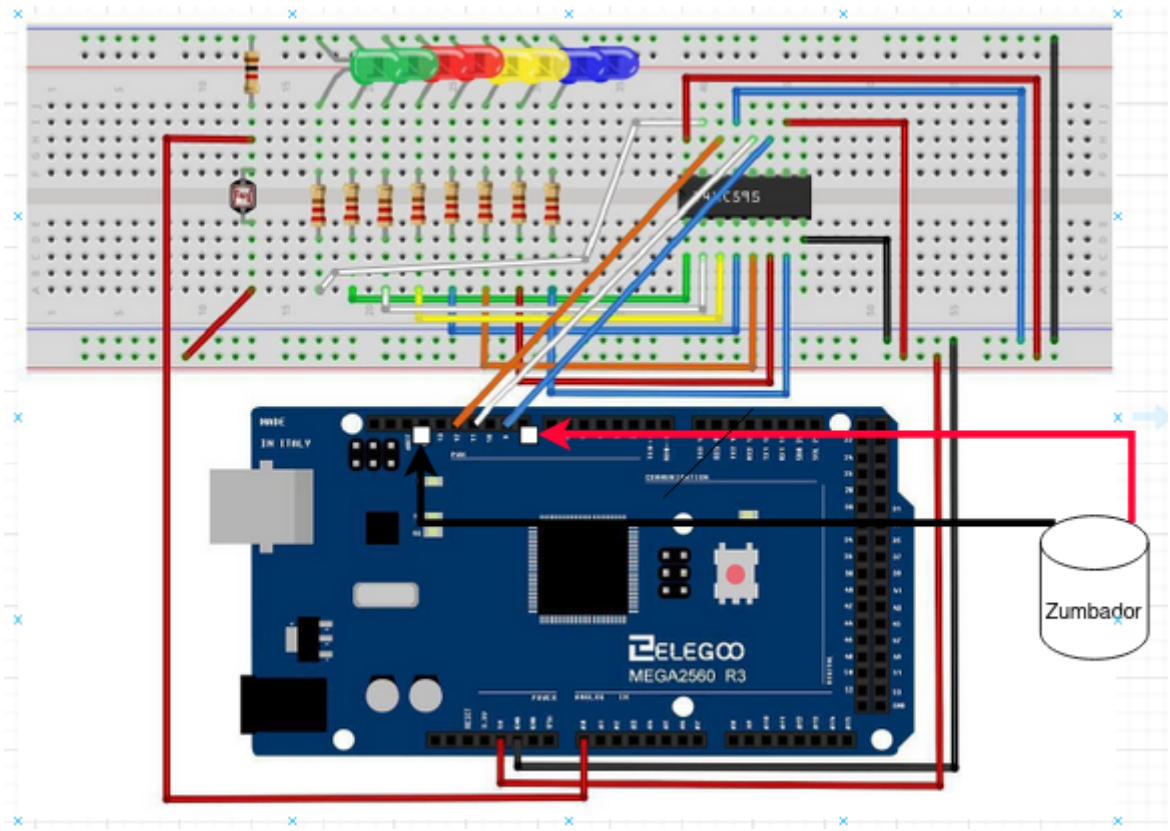
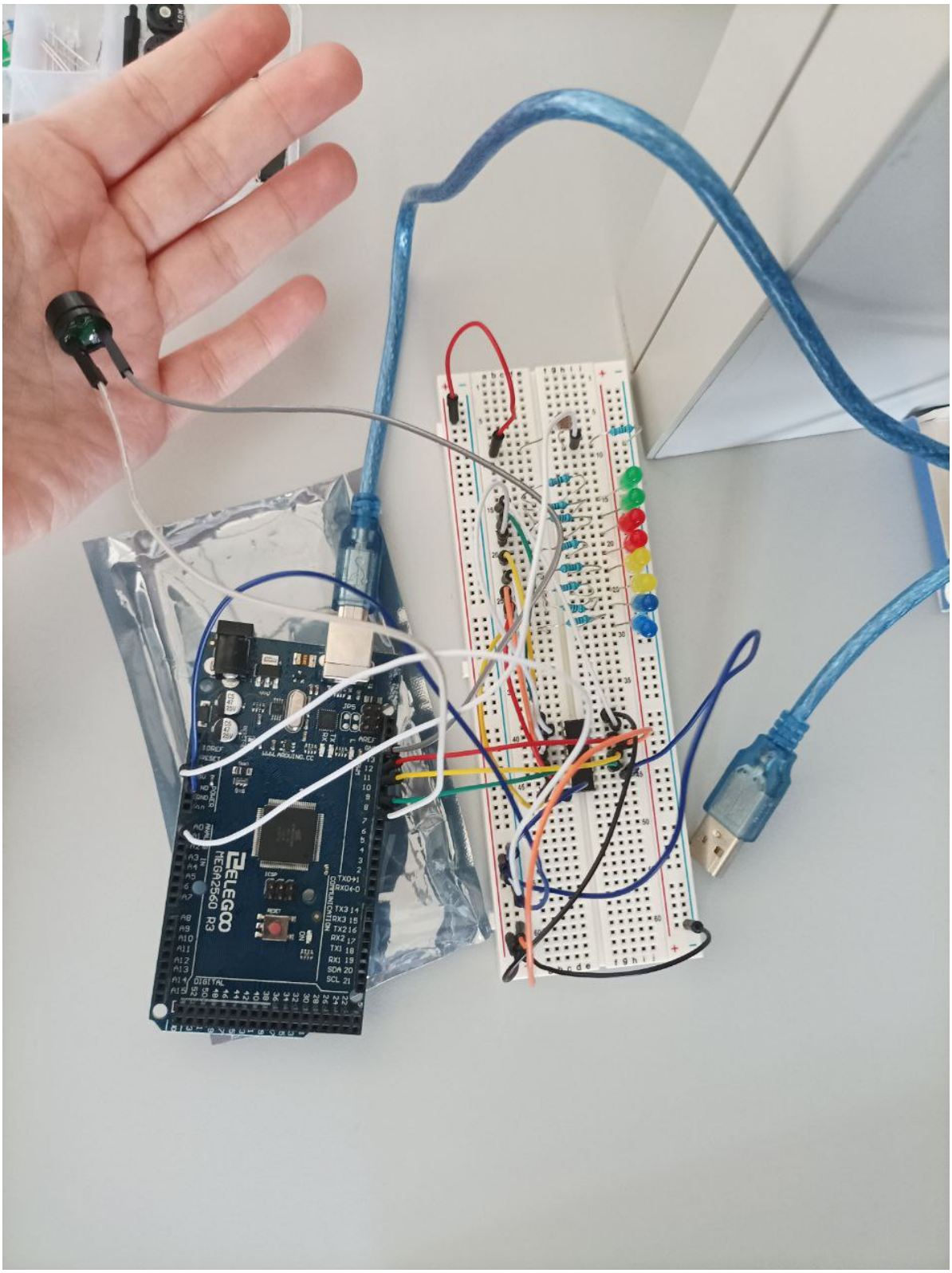


Foto:



Vídeo:

Compartido en Drive (acceso limitado a cuentas go.ugr):

https://drive.google.com/file/d/1E3XZNcWb4uTdhd5TA5qERfZ_d3pApzh_/view?usp=sharing