



UNIVERSIDAD DE GRANADA

ESTRUCTURA DE COMPUTADORES
GRADO EN INGENIERÍA INFORMÁTICA
CURSO 2018-2019



Memoria Práctica 5. E/S con Arduino.

Félix Ramírez García
felixramirezgarcia@correo.ugr.es

4 de diciembre de 2018

Índice

1	Introduccion	3
2	Hacer parpadear el led de la placa	3
3	Zumbador pasivo	3
4	Theremín de luz	4

Índice de figuras

3.1	Esquema zumbador pasivo	5
4.1	Esquema Theremin de luz	6
4.2	Montaje Theremin de luz en laboratirio	7

Índice de tablas

1. Introduccion

A continuación se van a mostrar varios ejemplos de uso de Elegoo , se mostrara para cada caso su código final, un esquema , una imagen y un breve vídeo con el programa funcionando.

Para descargar y ver los vídeos se proporciona un enlace a Consigna UGR, se ha subido el archivo con la durabilidad máxima posible, que es de un mes. Si para la corrección de la practica hubiese expirado este tiempo, póngase en contacto conmigo por el correo indicado en la primera pagina y le facilitaré un enlace actualizado . El enlace es el siguiente:

Para descargar vídeos pinchar aquí.

2. Hacer parpadear el led de la placa

El código que se ha cargado en la placa Elegoo para hacer parpadear la luz led de la placa ha sido el siguiente:

```
void setup(){
    pinMode(LED_BUILTIN, OUTPUT);
}

void loop(){
    digitalWrite(LED_BUILTIN, HIGH);
    delay(1000);
    digitalWrite(LED_BUILTIN, LOW);
    delay(1000);
}
```

En este caso el esquema eléctrico no es necesario, ya que simplemente se ha conectado la placa al ordenador por USB.

El vídeo correspondiente a este caso se encuentra en la carpeta comprimida dentro de Consigna (Enlace en la introducción), su nombre es led_placa.mp4

3. 5.1 Zumbador pasivo

El código que se ha cargado en la placa Elegoo para hacer un zumbador pasivo ha sido el siguiente:

```
#include "pitches.h"

// notes in the melody:
int melody[]={NOTE_D4,NOTE_E4,NOTE_E4,NOTE_F4,NOTE_D4,NOTE_E4,NOTE_E4,
    NOTE_F4,NOTE_C4,
```

```

        NOTE_D4,NOTE_F4,NOTE_G4,NOTE_C4,NOTE_D4,NOTE_F4
        ,NOTE_G4,NOTE_B4,NOTE_A4});

int noteDurations[] = {4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4};

void setup() {

}

void loop() {
    // iterate over the notes of the melody:
    for (int thisNote = 0; thisNote < 18; thisNote++) {
        // to calculate the note duration, take one second
        // divided by the note type.
        //e.g. quarter note = 1000 / 4, eighth note = 1000/8, etc
        .
        int noteDuration = 1000 / noteDurations[thisNote];
        tone(8, melody[thisNote], noteDuration);

        // to distinguish the notes, set a minimum time between them.
        // the note's duration + 30% seems to work well:
        int pauseBetweenNotes = noteDuration * 1.30;
        delay(pauseBetweenNotes);
        // stop the tone playing:
        noTone(8);
    }
}

```

El esquema eléctrico para hacer un zumbador pasivo es el que muestra la imagen 3.1.

El vídeo correspondiente a este caso se encuentra en la carpeta comprimida dentro de Consigna (Enlace en la introducción), su nombre es zumbador_pasivo.mp4 .

4. Theremín de luz

El código que se ha cargado en la placa Elegoo para hacer un Theremin de luz ha sido el siguiente:

```

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

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

```

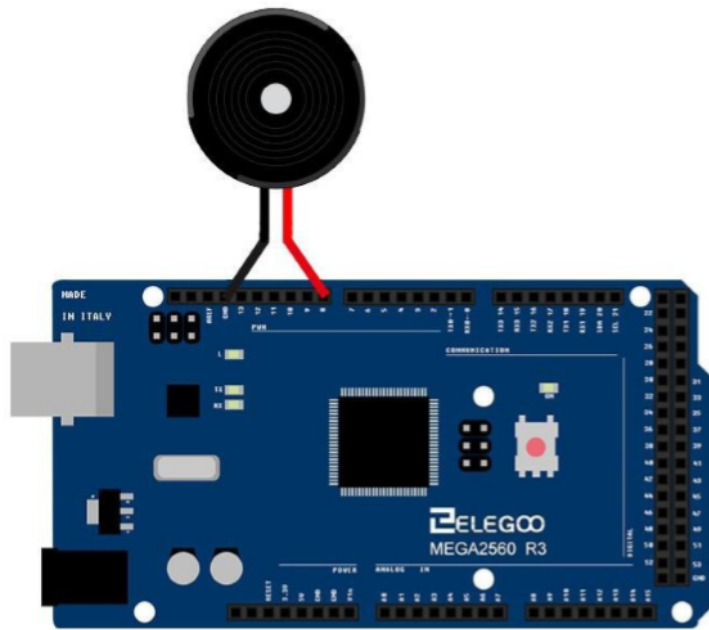


Figura 3.1: Esquema zumbador pasivo

```
// 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 loop() {
    //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, 20);  
  
// wait for a moment  
delay(10);  
}
```

El esquema eléctrico para hacer un un Theremin de luz es el que muestra la imagen 4.1 :

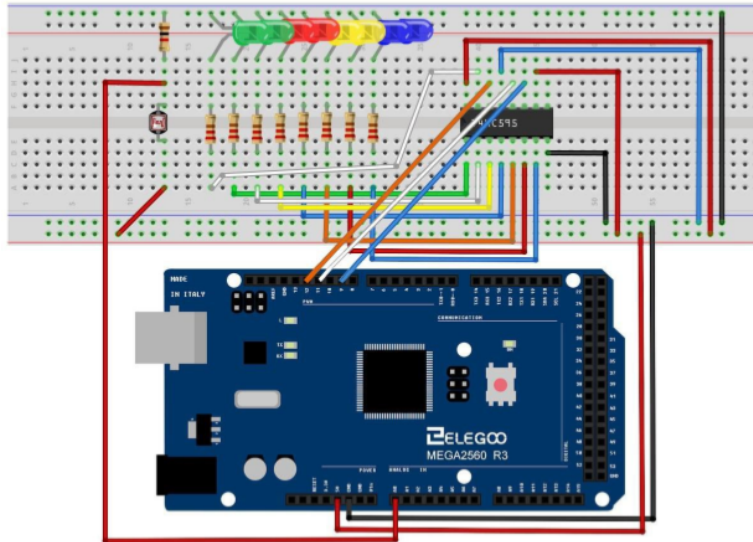


Figura 4.1: Esquema Theremin de luz

Y la figura 4.2 muestra una imagen con la instalación realizada en el laboratorio.

El vídeo correspondiente a este caso se encuentra en la carpeta comprimida dentro de Consigna (Enlace en la introducción), su nombre es theremin_luz.mp4

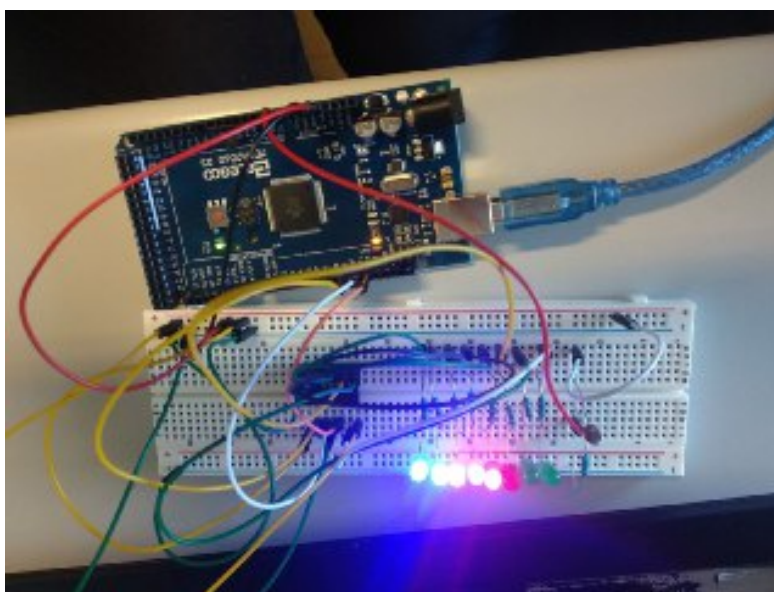


Figura 4.2: Montaje Theremin de luz en laboratorio