

Proyecto

Contador Regresivo con aumento de tiempo

Proyecto contador regresivo variable

1.¿De que se trata el Proyecto?.....	3
2.Nuestras Motivaciones y/o necesidades.....	3
3.Alcances del proyecto.....	3
4.Diseño de la estructura en 3D.....	3
Imagen lateral Imagen frontal.....	4
5.Diseño del Esquemático Eléctrico.....	4
6.Lista de Materiales.....	5
7.Diagrama de Flujo.....	5
8.Código de Programación.....	6
Se adjunta como fichero externo llamado sketch.ino.....	6
8.1 Variables.....	6
8.2 Cuerpo.....	7
9.Lista de testeos.....	8
9.1 Arrancar Arduino.....	8
9.2 Pulsar el botón de inicio.....	8
9.3 Pulsar el botón de suma.....	8
9.Conclusiones.....	8

1.¿De que se trata el Proyecto?

Nuestro proyecto trata un contador que realiza una cuenta atrás de 10 segundos por defecto cuando se pulsa un botón. Mediante otro botón, suma de 5 en 5 segundos para añadirlo a la cuenta regresiva, siempre y cuando la cuenta hacia atrás no este ya iniciada.

Una vez finalizada la cuenta atrás y llegue a 0, sonará un pitido tres veces intermitentemente y el contador volverá a su cuenta inicial, a la espera de volver iniciar el proceso.

2.Nuestras Motivaciones y/o necesidades

A la hora de disfrutar rodeado de amig@s o familia y de estar jugando a juegos de mesa, es posible que quizás algunas personas se demoren demasiado en realizar sus jugadas en un tiempo razonable. Es por ello que sale esta necesidad y nuestra motivación para no aburrir a los comensales y hacerles esperar demorando una partida más de lo necesario.

3.Alcances del proyecto

Conseguir un sistema robusto impreso en 3D para agilizar los turnos de las partidas en los juegos de mesa o en cualquier otra actividad que requiera una cuenta atrás, mediante un sistema sencillo pero efectivo de dos pulsadores.

4.Diseño de la estructura en 3D

Adjuntado en el proyecto los ficheros con extensión STL para su impresión.

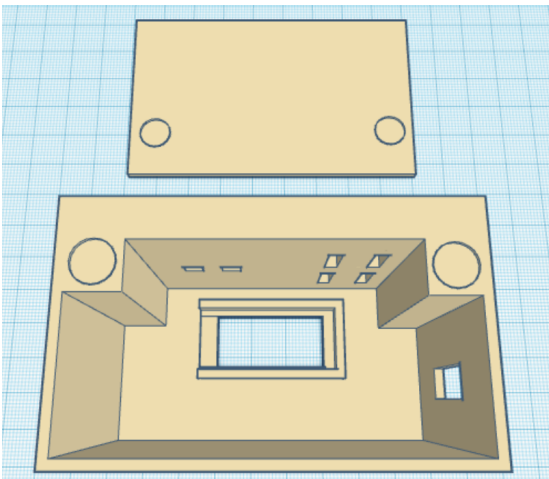


Imagen trasera

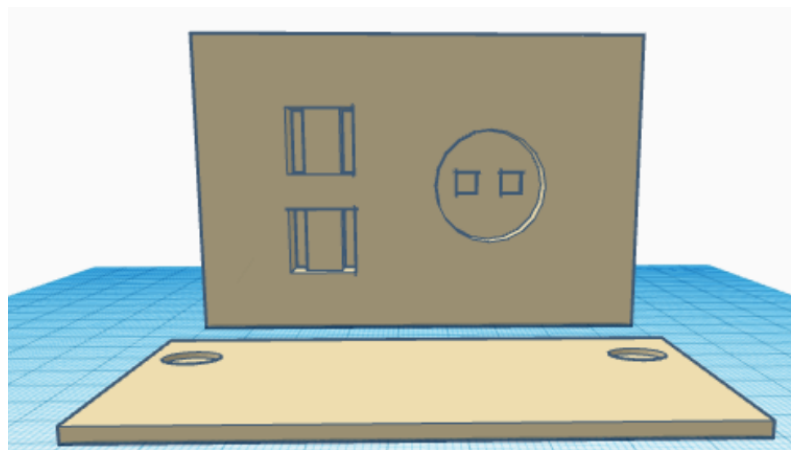


Imagen superior

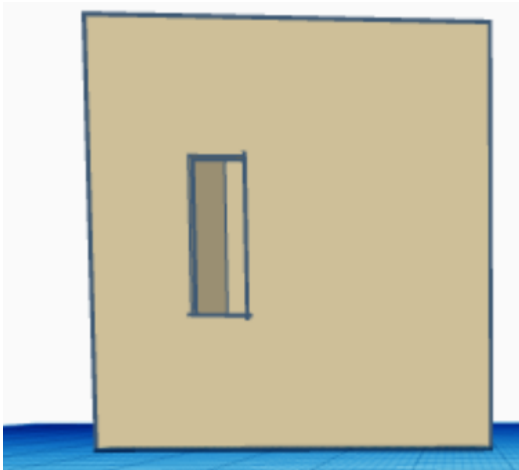


Imagen lateral

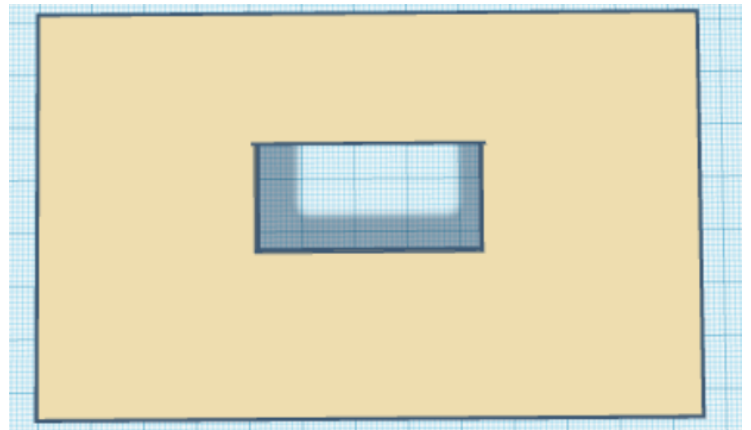


Imagen frontal

5. Diseño del Esquemático Eléctrico

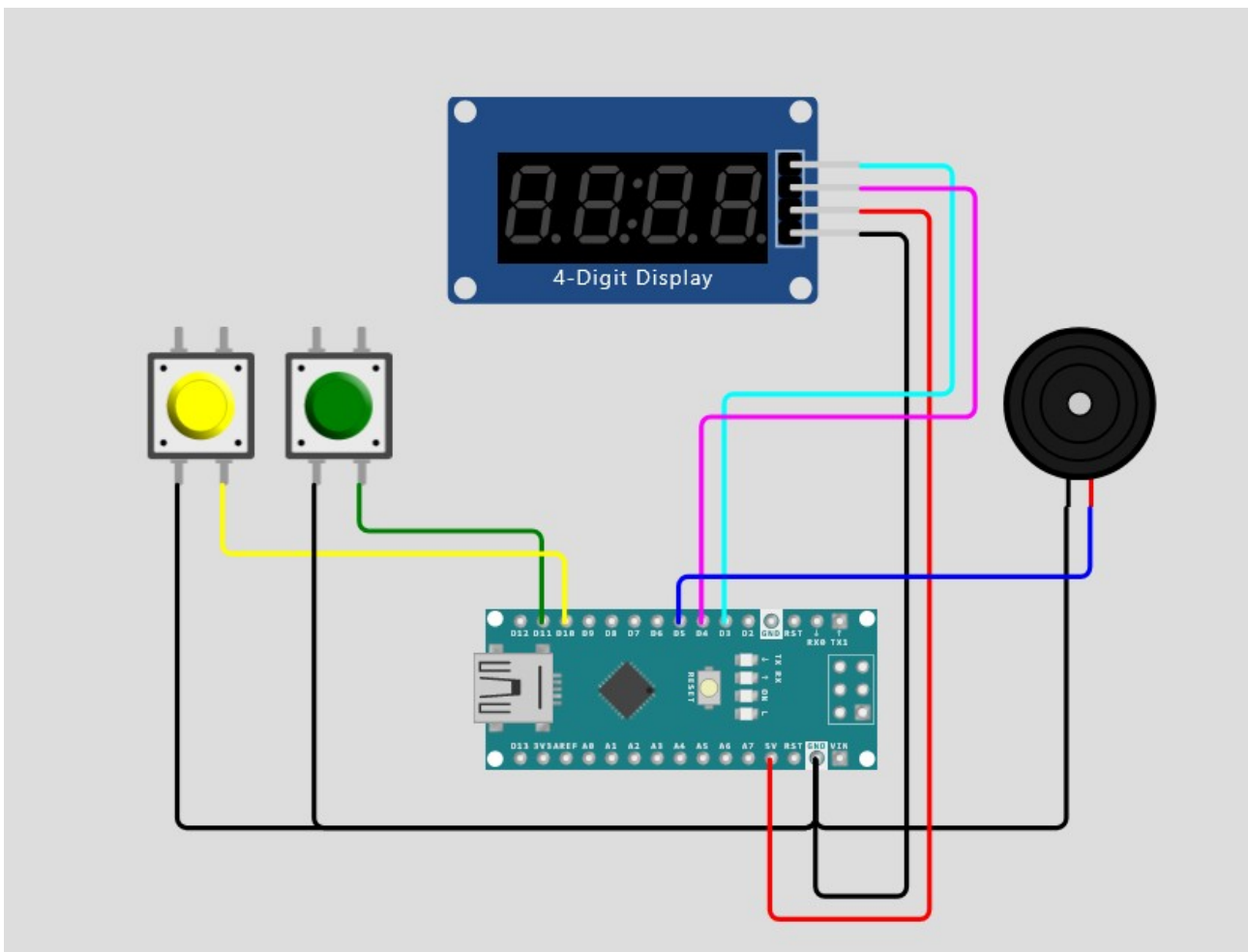


Imagen – Circuito Eléctrico

6.Lista de Materiales

Se han utilizado los siguientes materiales:

- Arduino NANO.
- Cables de conexiones.
 - Cables Dupont.
 - Cable regido con suficiente ancho de diámetro para realizar soldaduras.
- Dos pulsadores eléctrico.
- Protoboard.
- Cápsula piezo (PKM17EPP-4001-B0).
- 4 imanes de diámetro 0,8 mm y altura 1 mm.

7.Diagrama de Flujo

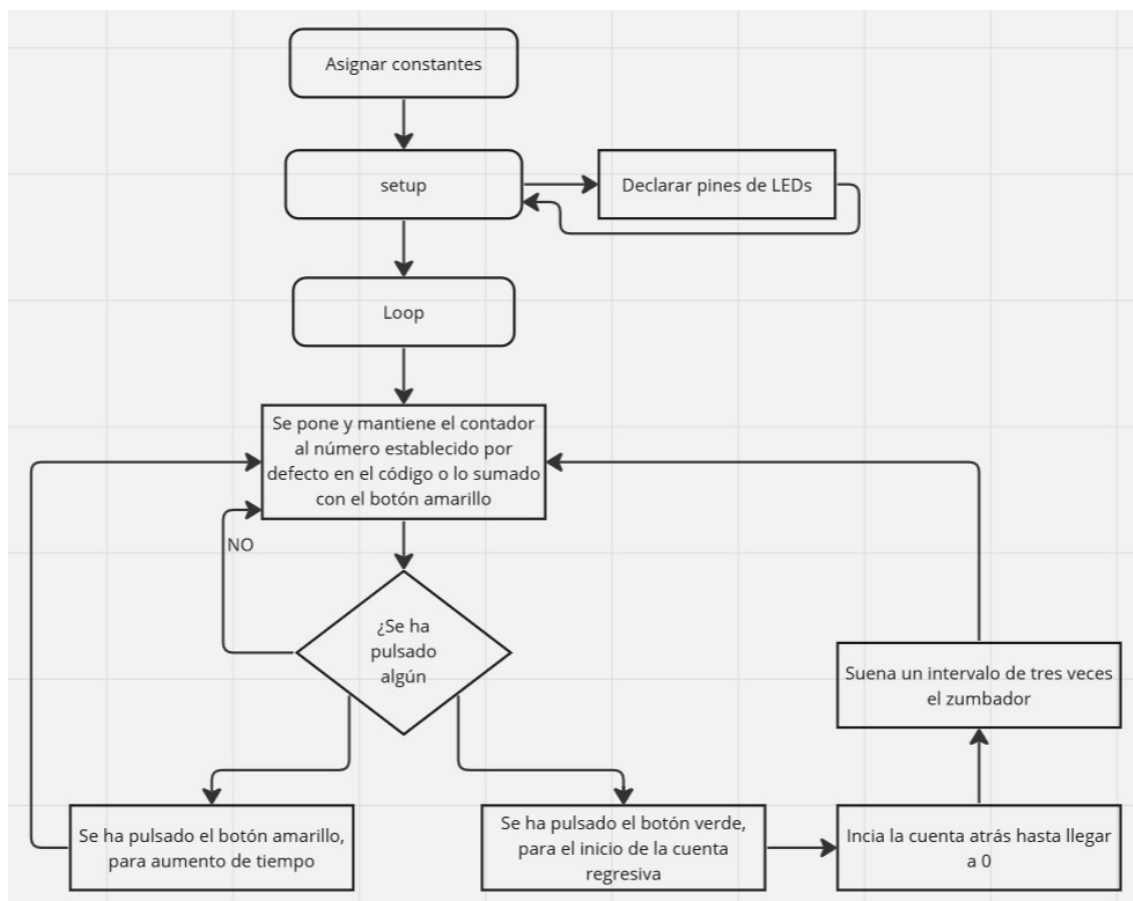


Imagen – Diagrama de flujo

8.Código de Programación

Se adjunta como fichero externo llamado sketch.ino.

8.1 Variables

```
#include <Config.h>
#include <EasyBuzzer.h>
#include <TM1637Display.h>

long tiempoContadorConstante = 10;
long tiempoContador = tiempoContadorConstante;
long tiempoIntervalo = 1000;
long tiempoAnterior = 0;
long tiempoActual = 0;
long tiempoMaximo = 25;
boolean iniciaPartida = false;
int valuePulsador = 0;
int frecuenciaHercios = 2000;

const int pulsador_verde = 11;
const int pulsador_amarillo = 10;
const byte inputBuzzer = 5;

#define CLK 3
#define DIO 4

// Crea objeto display
TM1637Display display = TM1637Display(CLK, DIO);
```

8.2 Cuerpo

```
void loop() {  
  if (digitalRead(pulsador_verde) == LOW) {  
    delay(100);  
    while (digitalRead(pulsador_verde) == LOW) {}  
    iniciaPartida = true;  
    tiempoContador = tiempoContadorConstante;  
  }  
  
  if (digitalRead(pulsador_amarillo) == LOW && iniciaPartida == false) {  
    delay(100);  
    while (digitalRead(pulsador_amarillo) == LOW) {}  
    if (tiempoContadorConstante < tiempoMaximo) {  
      tiempoContadorConstante = tiempoContadorConstante + 5;  
    }  
    muestraTiempo(tiempoContadorConstante);  
  }  
  
  if (tiempoContador >= 0 && iniciaPartida) {  
    tiempoActual = millis();  
    if ((tiempoActual - tiempoAnterior) >= tiempoIntervalo) {  
      muestraTiempo(tiempoContador);  
      tiempoContador--;  
      tiempoAnterior = tiempoActual;  
    }  
  }  
}  
  
void muestraTiempo(int tiempoContador) {  
  display.showNumberDec(tiempoContador);  
  if (tiempoContador == 0) {  
    tiempoContador = tiempoContadorConstante;  
    muestraTiempo(tiempoContador);  
    for (int i = 0; i <= 3; i++) {  
      tone(inputBuzzer, frecuenciaHercios);  
      delay(500);  
      noTone(inputBuzzer);  
      delay(800);  
    }  
    iniciaPartida = false;  
  }  
}
```

9.Lista de testeos

9.1 Arrancar Arduino

Al arrancar el sistema, vemos que efectivamente se pone el contador a 10 (Lo establecido por aplicación).

9.2 Pulsar el botón de inicio

Pulsar el botón de inicio y ver como realiza la cuenta atrás e intentar sumar 5 segundos mas al contador, y efectivamente no suma, ya que no se realiza hasta que finalice el proceso. Una vez finalizado vemos como pita tres veces y vuelve al contador inicial.

9.3 Pulsar el botón de suma

Antes de iniciar el proceso de cuenta atrás, pulsamos el botón para la suma de 5 segundos adicionales al contador. Efectivamente los suma e iniciamos el testeo con el nuevo tiempo pulsando el otro botón de inicio para comprobar que el flujo es correcto.

9.Conclusiones

Durante el ensamblaje y la construcción se pensó que además de realizar un sumatorio mediante un pulsador, también podría existir un pulsador de resta. E incluso simplificar ambos pulsadores con un potenciómetro.

El diseño de la caja impresa en 3D es bastante reducida, con lo que se recomienda utilizar cables más cortos en las construcción para evitar una maraña de cables innecesaria.

No obstante, cumple su función esperada.