**Semáforo con pase de peatones Arduino**

Se trata de un mini proyecto que hice para probar y aprender a programar Arduino.

Este semáforo no cambia por periodos de tiempo, cambia según como se interactúe con el.

El ciclo empieza con el semáforo para peatones en verde con el Buzzer pitando intermitentemente para que los peatones invidentes sepan que pueden pasar mientras que el semáforo de coches se encuentra en rojo. Para que el semáforo de coches cambie se usa el sensor de ultrasonido que mide la distancia en un área para saber si existen coches detenidos esperando para ello también se tiene que cumplir una regla de 6 segundos seguidos a menos de esa distancia. En este caso y como es un ejemplo puse que si la distancia es menor a 100 centímetros durante 6 segundos cambiara. Una vez se encuentre en verde el semáforo de coches y en rojo el de peatones para que cambie otra vez el peatón tendrá que pulsar el botón.

Los procesos de cambios no son inmediatos una vez accionados tardan 5 segundos en cambiar de verde a rojo además entre los cambios de semáforo a semáforo se aplica un segundo y medio mas.

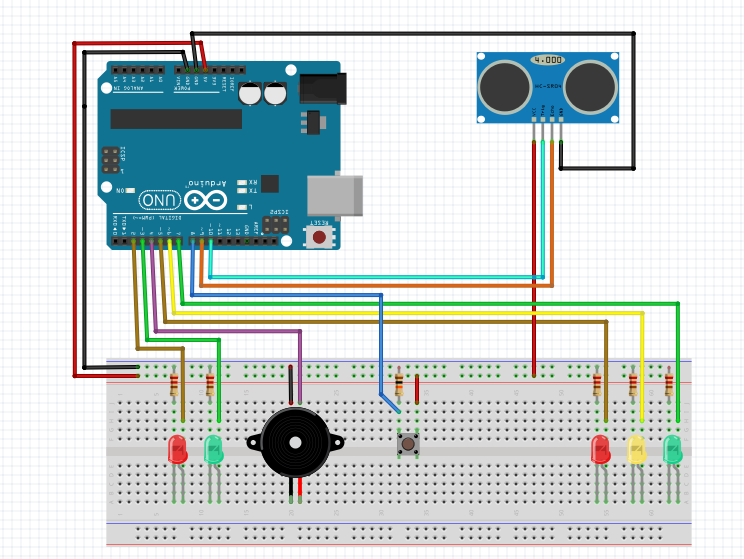
Proceso de cambio:

Peatones

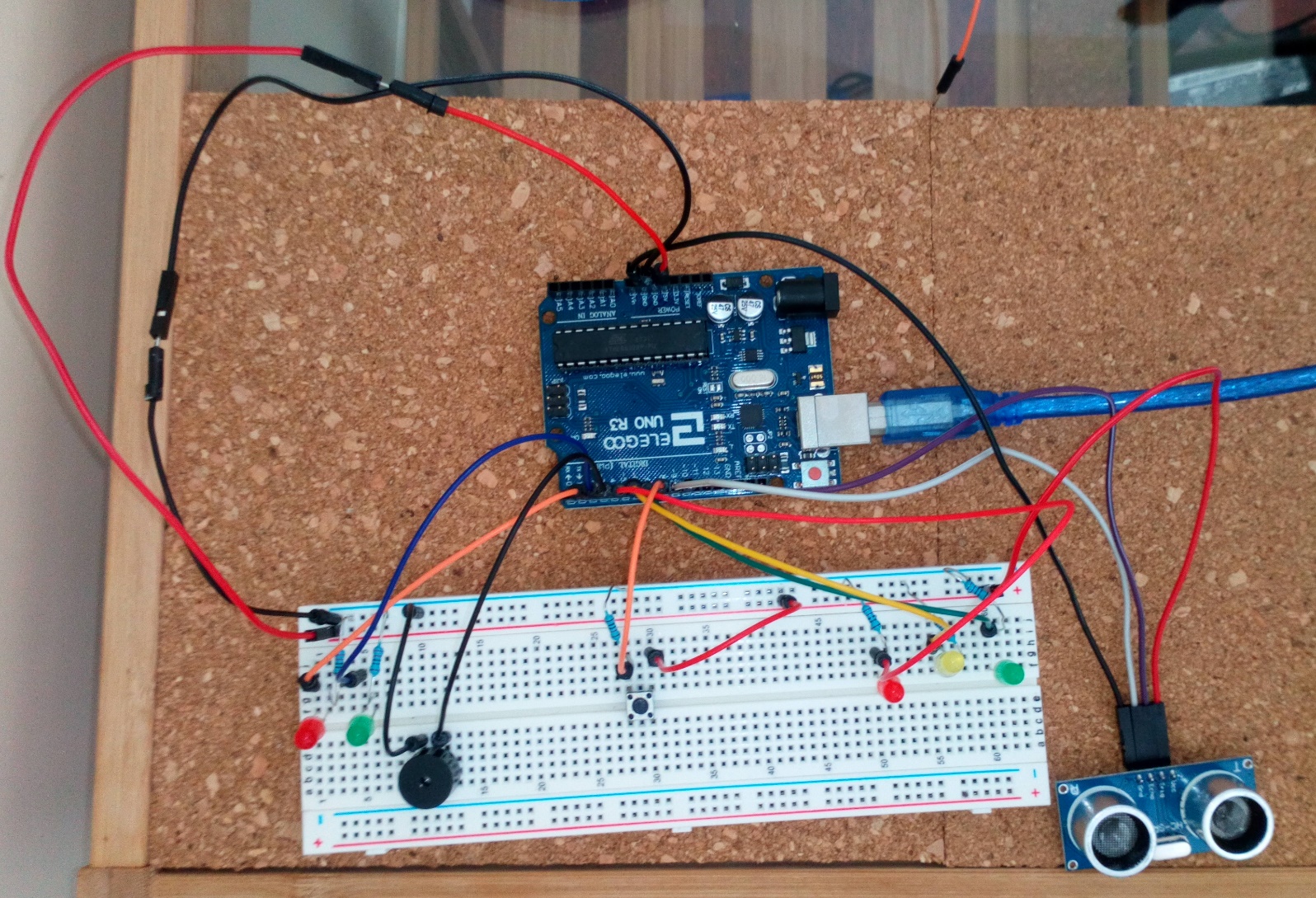
**Componentes**

* Placa de Elegoo uno R3
* Sensor de ultrasonido HC-SR04
* Botón (Pequeño)
* Buzzer pasivo
* 5 Led
* 5 resistencias 220 ohmios
* 1 resistencia 10k ohmios

**Plano**



En plano se ve muy bonito en la realidad es algo como esto



Código:

#include <SR04.h>

// Nombre de colores asociados con la salidas digitales correspondiente.

#define PEATON\_LED\_ROJO 2

#define PEATON\_LED\_vERDE 3

#define PEATON\_BUZZER 4

#define COCHE\_LED\_ROJO 5

#define COCHE\_LED\_AMARILLO 6

#define COCHE\_LED\_VERDE 7

#define BOTTON\_PEATON 8

#define ECHO\_PIN 9

#define TRIG\_PIN 10

bool bPasoPeatonesActivo = true;

int iTiempoCambioLed = 5000;

int iTiempoEntreSemaforo = 1500;

int iTiempoCambioSemaforo = 5000;

int iPulsadorPeatones;

int iPulsadorCoches;

int iIteradorCoches;

int iBottonEmergencia;

long a;

//SENSOR SR04

SR04 sr04 = SR04(ECHO\_PIN,TRIG\_PIN);

void setup() {

Serial.begin(9600);

// Definir funcionamiento de cada PIN

pinMode(PEATON\_LED\_ROJO, OUTPUT);

pinMode(PEATON\_LED\_vERDE, OUTPUT);

pinMode(PEATON\_BUZZER, OUTPUT);

pinMode(COCHE\_LED\_ROJO, OUTPUT);

pinMode(COCHE\_LED\_AMARILLO, OUTPUT);

pinMode(COCHE\_LED\_VERDE, OUTPUT);

pinMode(BOTTON\_PEATON, INPUT);

// Estado inicial: semaforo de peatones en verde y coches en rojo

digitalWrite(PEATON\_LED\_ROJO, LOW);

digitalWrite(PEATON\_LED\_vERDE, HIGH);

digitalWrite(COCHE\_LED\_ROJO, HIGH);

digitalWrite(COCHE\_LED\_AMARILLO, LOW);

digitalWrite(COCHE\_LED\_VERDE, LOW);

bPasoPeatonesActivo = true;

}

void loop() {

if (bPasoPeatonesActivo){

sensorProximidad();

semaforoPeatones();

} else {

iPulsadorPeatones = digitalRead(BOTTON\_PEATON);

if (iPulsadorPeatones == HIGH) {

semaforoCoches();

bPasoPeatonesActivo = true;

}

}

}

void semaforoPeatones() {

delay(iTiempoCambioSemaforo-3000);

for (int i = 0; i <= 6; i++) {

digitalWrite(PEATON\_LED\_vERDE, LOW);

delay(250);

digitalWrite(PEATON\_LED\_vERDE, HIGH);

delay(250);

sonarBuzzer("CORTO");

}

digitalWrite(PEATON\_LED\_vERDE, LOW);

digitalWrite(PEATON\_LED\_ROJO, HIGH);

delay(iTiempoEntreSemaforo);

digitalWrite(COCHE\_LED\_ROJO, LOW);

digitalWrite(COCHE\_LED\_VERDE, HIGH);

bPasoPeatonesActivo = false;

}

void semaforoCoches() {

delay(iTiempoCambioSemaforo);

digitalWrite(COCHE\_LED\_VERDE, LOW);

digitalWrite(COCHE\_LED\_AMARILLO, HIGH);

delay(iTiempoCambioLed);

digitalWrite(COCHE\_LED\_AMARILLO, LOW);

digitalWrite(COCHE\_LED\_ROJO, HIGH);

delay(iTiempoEntreSemaforo);

digitalWrite(PEATON\_LED\_ROJO, LOW);

digitalWrite(PEATON\_LED\_vERDE, HIGH);

bPasoPeatonesActivo = true;

sonarBuzzer("LARGO");

}

void sonarBuzzer(String sVelocidad) {

if (sVelocidad == "CORTO") {

for (int i = 0; i <= 8; i++) {

tone(PEATON\_BUZZER, 500);

delay(50);

noTone(PEATON\_BUZZER);

delay(50);

}

} else { sensorProximidad(); }

}

void sensorProximidad() {

int iX = 1;

while (iX==1){

a=sr04.Distance();

delay(400);

tone(PEATON\_BUZZER, 500);

delay(150);

noTone(PEATON\_BUZZER);

delay(150);

if ((a <= 100) and (a > 0 )) {

iIteradorCoches++;

} else { iIteradorCoches = 0; }

if (iIteradorCoches >= 6) {

iIteradorCoches = 0;

iX=2;

}

}

}