

MATERIA

Sistemas Programables

CARRERA

Ingeniería en sistemas Computacionales

PRESENTA:

Jorge Alberto Sanchez Maldonado

NOMBRE DE LA MAESTRA:

Ing. Carlos Rafael Levy Rojas

LEÓN, GUANAJUATO

Periodo: Enero-Junio 2018

Código

```
/******  
  DEFINICIÓN DE CONSTANTES  
******/  
  
// Definición de pines  
const int led_verde_auto = 13;  
const int led_ambar_auto = 12;  
const int led_rojo_auto = 11;  
const int led_verde_peaton = 10;  
const int led_rojo_peaton = 9;  
const int pin_pulsador = 8;  
  
// Tiempo mínimo que permanecerá en verde el semáforo para automóviles  
const int tiempo_minimo_verde_auto = 7000;  
  
/******  
  FUNCIONES AUXILIARES DEL PROGRAMA  
******/  
  
//  
// Inicializa semáforo. Verde para automóviles  
//  
void semaforo_init(void) {  
  // Enciende el led verde para automóviles. Apaga los demás leds  
  digitalWrite(led_verde_auto, HIGH);  
  digitalWrite(led_rojo_auto, LOW);  
  digitalWrite(led_ambar_auto, LOW);  
  
  // Enciende el led rojo para peatones. Apaga los demás leds  
  digitalWrite(led_rojo_peaton, HIGH);  
  digitalWrite(led_verde_peaton, LOW);  
}  
  
//  
// Cambia el semáforo de automóviles a rojo  
//  
void semaforo_auto_rojo(void) {  
  
  // Apaga el led verde para coches  
  digitalWrite(led_verde_auto, LOW);  
  
  // Enciende el led ambar para coches y parpadea 3 veces  
  for(int i=4; i>0; i--) {  
    delay(500);
```

```

        digitalWrite(led_ambar_auto, HIGH);
        delay(500);
        digitalWrite(led_ambar_auto, LOW);
    }

    // Enciende el led rojo para coches
    digitalWrite(led_rojo_auto, HIGH);
}

//
// Espera a que se presione el pulsador
//
void semaforo_espera_pulsador(void) {
    int presionado;
    long time;

    // Espera a que se presione el pulsador
    // Debe permanecer esperando un tiempo mínimo
    time = millis() + tiempo_minimo_verde_auto;
    presionado = 0;
    while(1) {
        // Lee el estado del pulsador
        if (digitalRead(pin_pulsador) == LOW)
            presionado = 1;

        // Si ha pasado el tiempo de verde para coches
        // y se ha presionado el pulsador, salir
        if ((millis() > time) && (presionado == 1))
            break;
    }
}

//
// Espera un tiempo que depende del potenciómetro
//
void semaforo_delay(void) {
    int tiempo_peatones;

    // Calcula el tiempo de espera.
    // 2 segundos más el tiempo que añada el potenciómetro
    tiempo_peatones = 2000 + 5 * analogRead(A0);

    // Envía por el puerto serie el tiempo de espera
    Serial.print("Tiempo peatones=");
    Serial.println(tiempo_peatones);
}

```

```

    // Espera
    delay(tiempo_peaton);
}

//
// Cambia el semáforo de peatones a rojo
//
void semaforo_peaton_rojo(void) {

    // Parpadea el led verde 3 veces
    for(int i=3; i>0; i--) {
        digitalWrite(led_verde_peaton, LOW);
        delay(500);
        digitalWrite(led_verde_peaton, HIGH);
        delay(500);
    }

    // Enciende el led rojo para peatones y apaga el verde
    digitalWrite(led_verde_peaton, LOW);
    digitalWrite(led_rojo_peaton, HIGH);
}

/*****
  FUNCIONES PRINCIPALES DEL PROGRAMA
*****/

//
// Inicialización del programa
//
void setup() {
    // Define las salidas de Arduino
    pinMode(led_verde_auto, OUTPUT);
    pinMode(led_ambar_auto, OUTPUT);
    pinMode(led_rojo_auto, OUTPUT);
    pinMode(led_verde_peaton, OUTPUT);
    pinMode(led_rojo_peaton, OUTPUT);

    // Define las entradas con resistencia de Pull-up
    pinMode(pin_pulsador, INPUT_PULLUP);

    // Inicializa el puerto de comunicaciones
    Serial.begin(9600);
    Serial.println("Semaforo en marcha");
}

```

```
//  
// Programa principal  
//  
void loop() {  
  
    // Inicializa semáforo. Verde para automóviles  
    semaforo_init();  
  
    // Espera a que se presione el pulsador  
    semaforo_espera_pulsador();  
  
    // Cambia el semáforo de automóviles a rojo  
    semaforo_auto_rojo();  
  
    // Espera antes de permitir paso a peatones  
    delay(1500);  
  
    // Enciende el led verde para peatones  
    digitalWrite(led_rojo_peaton, LOW);  
    digitalWrite(led_verde_peaton, HIGH);  
  
    // Espera un tiempo que depende del potenciómetro  
    semaforo_delay();  
  
    // Cambia el semáforo de peatones a rojo  
    semaforo_peatones_rojo();  
  
    // Espera antes de permitir el paso a los automóviles  
    delay(2000);  
  
    // Comienza un nuevo ciclo  
}
```

Imágenes





