



TECNOLÓGICO NACIONAL DE MÉXICO

Instituto Tecnológico de León

Ingeniería en Sistemas Computacionales

“Sistemas programables”

Ing. Levy Rojas Carlos Rafael

Moreno Ciénega Cesar Omar

Practica 1

Semestre: Ene-Ago/2018.

20 de Febrero del 2010

Código

Archivo Editar Programa Herramientas Ayuda

```
practica_3_SP

int tiempo = 1000; //variable para controlar la velocidad de encendido y apagado de los LED's
int boton = 0; //Variable para guardar el estado del boton

void setup() {
    // initialize digital pin LED_BUILTIN as an output.
    pinMode(8, OUTPUT); // Salida del led verde de los coches
    pinMode(9, OUTPUT); // Salida del led amarillo de los coches
    pinMode(10, OUTPUT); // Salida del led rojo de los coches
    pinMode(11, OUTPUT); // Salida del led verde de los peatones
    pinMode(12, OUTPUT); // Salida del led rojo de los peatones
    pinMode(7, INPUT); //Entrada del boton
}

void parpadeo(int pin){ //método para hacer parpadear un LED (especificado en la variable pin).
    for(int i = 0; i < 3; i++){ //ciclo para hacer que el LED parpadee 3 veces
        digitalWrite(pin, LOW); //Apaga el LED pin
        delay(tiempo/3); //realiza una pausa
        digitalWrite(pin, HIGH); //Enciende el LED pin
        delay(tiempo/3); //realiza una pausa
    }
}

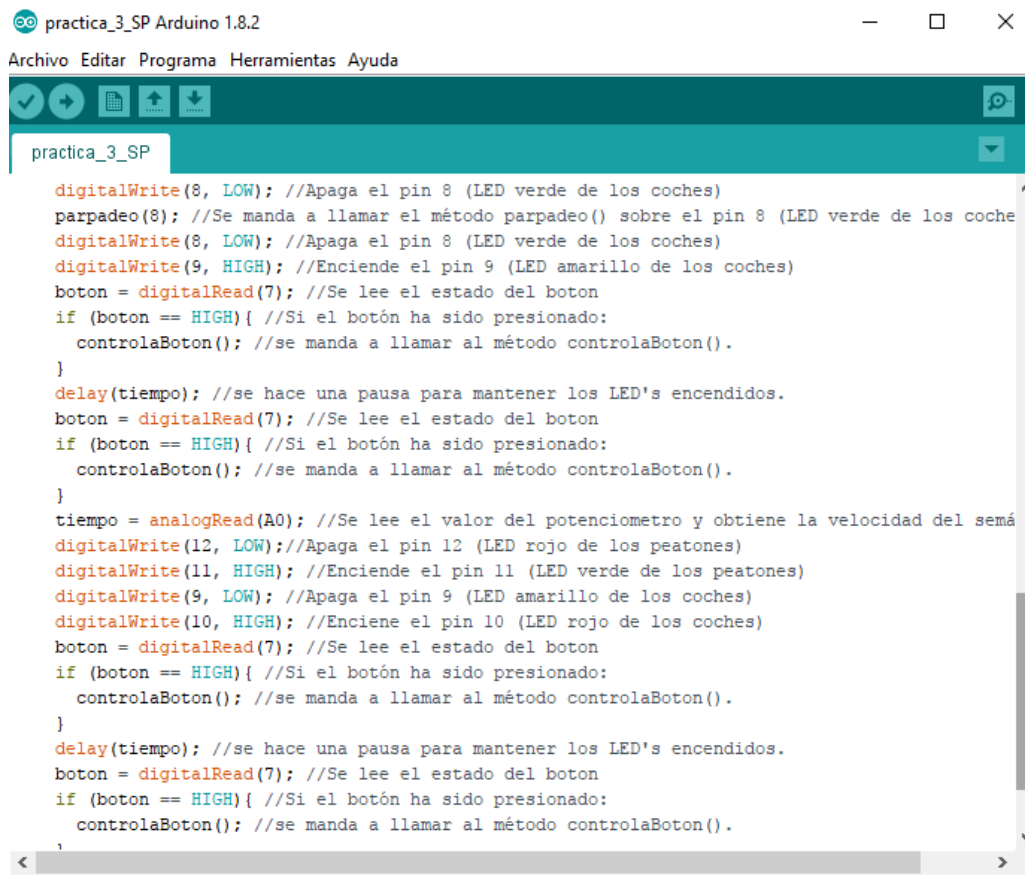
void controlaBoton(){ //método que realiza la acción del boton al ser presionado.
    digitalWrite(8, LOW); //apaga el LED verde de los coches
    digitalWrite(9, LOW); //apaga el LED amarillo de los coches
    digitalWrite(10, HIGH); //enciende el LED rojo de los coches
}
```

Archivo Editar Programa Herramientas Ayuda

```
practica_3_SP

digitalWrite(11, HIGH); //enciende el LED verde de los peatones
digitalWrite(12, LOW); //apaga el LED rojo de los peatones
delay(tiempo); //realiza una pausa
parpadeo(11); //llamada al método de parpadeo
}

// the loop function runs over and over again forever
void loop() {
    tiempo = analogRead(A0); //Se lee el valor del potenciómetro y obtiene la velocidad del semá
    boton = digitalRead(7); //Se lee el estado del boton
    if (boton == HIGH){ //Si el botón ha sido presionado:
        controlaBoton(); //se manda a llamar al método controlaBoton().
    }
    digitalWrite(10, LOW); //Después de evaluar el IF,
    digitalWrite(11, LOW); //apaga los pines 10 y 11 (LED verde de peatones y LED rojo de coches)
    digitalWrite(8, HIGH); //Enciende el pin 8 (LED verde de los coches)
    digitalWrite(12, HIGH); //Al mismo tiempo, el pin 12 se enciende (el LED rojo de los peatones)
    boton = digitalRead(7); //Se lee el estado del boton
    if (boton == HIGH){ //Si el botón ha sido presionado:
        controlaBoton(); //se manda a llamar al método controlaBoton().
    }
    delay(tiempo*2); //se hace una pausa para mantener los LED's encendidos.
    boton = digitalRead(7); //Se lee el estado del boton
    if (boton == HIGH){ //Si el botón ha sido presionado:
        controlaBoton(); //se manda a llamar al método controlaBoton().
    }
    tiempo = analogRead(A0); //Se lee el valor del potenciómetro y obtiene la velocidad del semá
```

The image shows the Arduino IDE interface with the file 'practica_3_SP' open. The code is written in C++ and controls an LED traffic light system. It includes functions for turning LEDs on and off, reading a button, and controlling the timing of the lights based on a potentiometer's value. The code is as follows:

```
digitalWrite(8, LOW); //Apaga el pin 8 (LED verde de los coches)
parpadeo(8); //Se manda a llamar el método parpadeo() sobre el pin 8 (LED verde de los coches)
digitalWrite(8, LOW); //Apaga el pin 8 (LED verde de los coches)
digitalWrite(9, HIGH); //Enciende el pin 9 (LED amarillo de los coches)
boton = digitalRead(7); //Se lee el estado del boton
if (boton == HIGH){ //Si el botón ha sido presionado:
    controlaBoton(); //se manda a llamar al método controlaBoton().
}
delay(tiempo); //se hace una pausa para mantener los LED's encendidos.
boton = digitalRead(7); //Se lee el estado del boton
if (boton == HIGH){ //Si el botón ha sido presionado:
    controlaBoton(); //se manda a llamar al método controlaBoton().
}
tiempo = analogRead(A0); //Se lee el valor del potenciómetro y obtiene la velocidad del semá
digitalWrite(12, LOW); //Apaga el pin 12 (LED rojo de los peatones)
digitalWrite(11, HIGH); //Enciende el pin 11 (LED verde de los peatones)
digitalWrite(9, LOW); //Apaga el pin 9 (LED amarillo de los coches)
digitalWrite(10, HIGH); //Enciende el pin 10 (LED rojo de los coches)
boton = digitalRead(7); //Se lee el estado del boton
if (boton == HIGH){ //Si el botón ha sido presionado:
    controlaBoton(); //se manda a llamar al método controlaBoton().
}
delay(tiempo); //se hace una pausa para mantener los LED's encendidos.
boton = digitalRead(7); //Se lee el estado del boton
if (boton == HIGH){ //Si el botón ha sido presionado:
    controlaBoton(); //se manda a llamar al método controlaBoton().
}
```

Explicación

El circuito representa un semáforo peatonal y un semáforo para automóviles.

Consta de cinco led's tres de esos led's son para los automóviles y dos de esos led's para los peatones, consta de un push boton que al momento de presionarlo se activa el semaforo del peaton en verde, y el de automóvil en rojo, después de un tiempo vuelve el ciclo a la normalidad y así sucesivamente cada que se presione el boton, también consta de un potenciómetro, el cual sirve para la frecuencia en la cual se lleva el ciclo.

Foto evidencia

