

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
Facultad de Ingeniería, Arquitectura y Diseño



LABORATORIO DE SISTEMAS CON MICROCONTROLADOR

Lugar y fecha de elaboración:
Ensenada, Baja California a 30 de agosto de 2021

Práctica No. 03:
Diseño de un semáforo digital

5to Semestre
Ingeniería en Electrónica - Grupo 352

Elaborado por:
Muñiz Hernández Oscar Javier - 358851

Profesor:
Dr. Everardo Inzunza González

Objetivo

Desarrollar un sistema basado en Microcontrolador Arduino para controlar un semáforo de cruce de una calle principal con una avenida y que considere el cruce de peatones.

Requerimientos generales del sistema

Se requiere diseñar un semáforo digital para controlar el tráfico vehicular en la intersección de una calle de tráfico muy denso con una calle de tráfico moderado.

La calle principal deberá tener una luz verde encendida durante un tiempo de 15 segundos. Antes de agotarse el tiempo de 15 segundos, la luz verde deberá parpadear 3 veces y posteriormente apagarse. La calle lateral debe tener la luz verde encendida durante 10 segundos, de igual manera antes de agotarse los 10 segundos la luz verde deberá parpadear 3 veces. La luz ámbar de precaución de ambos semáforos tiene que durar 3 segundos y es activada cuando se apaga la luz verde de cada semáforo. La luz roja se deben mantener encendidas mientras la luz verde o ámbar del semáforo contrario están encendidas.

Lista de material y equipo

- 1 Microcontrolador Arduino UNO
- 6 Optoacopladores MOC3011 o equivalente
- 6 TRIACs 2N6344 o equivalente
- 2 focos verdes, 2 focos ámbar (amarillos) y 2 focos rojos de 110 Vca @ 40Watts o menor..
- 6 sockets para focos de 110 Vca.
- 1 Clavija
- 3 mts Cable duplex AWG 14
- 12 R de 220Ω.
- 1 Computadora personal o laptop
- 1 software Arduino IDE

Antecedentes relacionados con la práctica

Arduino es una plataforma electrónica de código abierto basada en hardware y software fácil de usar. Las placas Arduino pueden leer entradas y convertirla en una salida, activando un motor, encendiendo un LED, publicando algo en línea. Puede decirle a su tablero qué hacer enviando un conjunto de instrucciones al microcontrolador en el tablero. Para hacerlo, utiliza el lenguaje de programación Arduino y el Software Arduino, basado en Processing.

Gracias a su experiencia de usuario simple y accesible, Arduino se ha utilizado en miles de proyectos y aplicaciones diferentes. El software Arduino es fácil de usar para principiantes, pero lo suficientemente flexible para usuarios avanzados. Funciona en Mac, Windows y Linux. Los profesores y los estudiantes lo utilizan para construir instrumentos científicos de bajo costo, para probar los principios de la química y la física, o para comenzar con la programación y la robótica.

Desarrollo experimental

La práctica se realiza primeramente construyendo el circuito de la figura 1 en el simulador PROTEUS, como son dos semáforos y cada semáforo requiere de 3 focos (verde, ámbar y rojo) se requerirán de 6 circuitos para el total del circuito ya que 1 moc y 1 triac es para 1 solo foco.

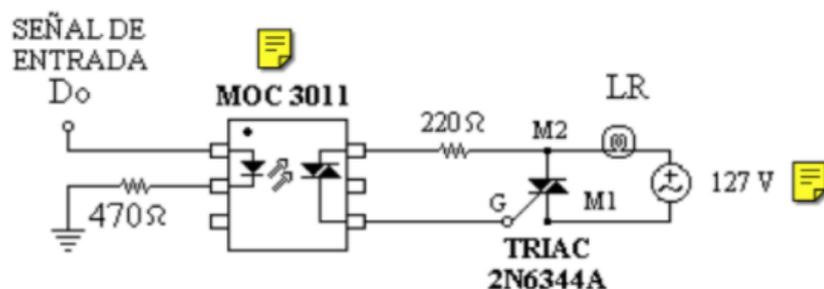


Figura 1. Circuito general el cual muestra los principales componentes el MOC y el TRIAC.

El circuito en PROTEUS me quedo de la siguiente manera:

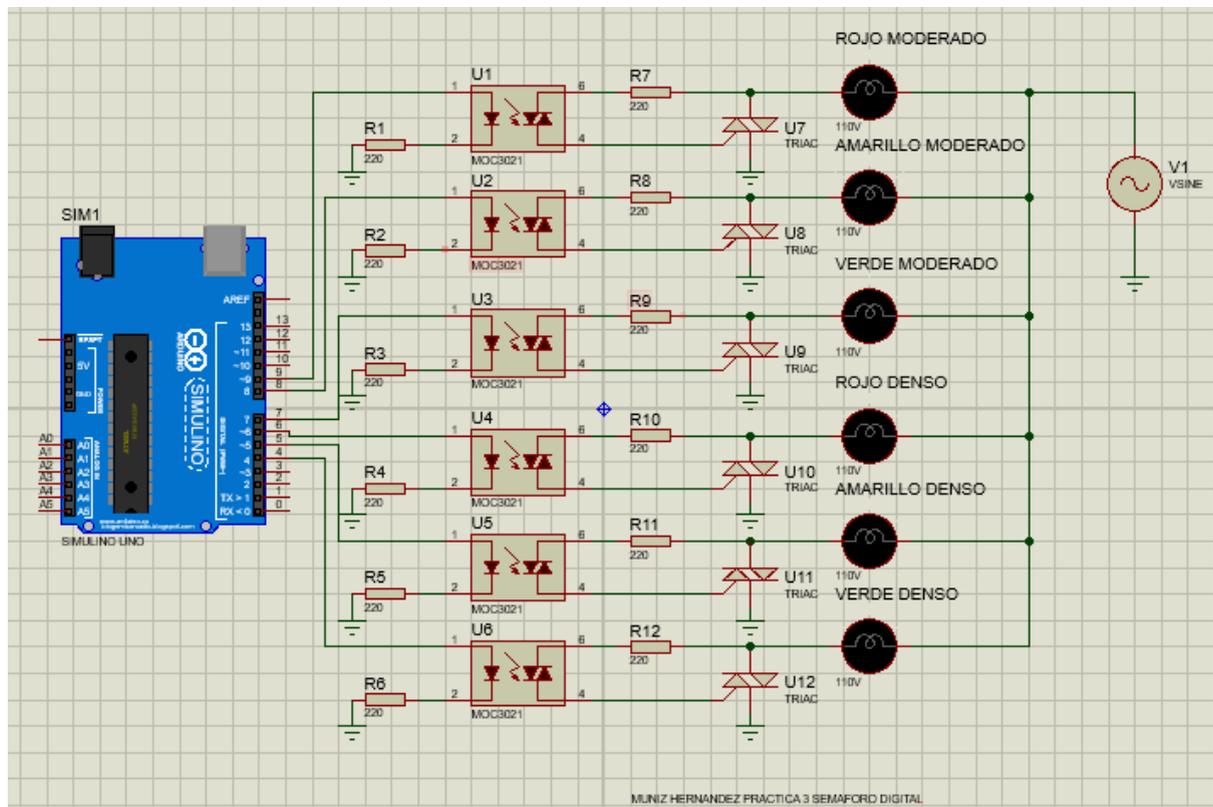


Figura 2. Circuito en el simulador utilizando 6 focos, 2 para cada color (verde, ámbar, rojo).

En la imagen se observa que están conectados a la misma tierra porque si lo pongo de otra manera no funciona.

Después se realizó el mismo procedimiento pero ahora en físico, para ello se necesito la protoboard, además de los materiales requeridos.

Una vez armado el circuito se le cargó el código ya programado previamente al arduino, solo para llegar a conectarlo.

A continuación se muestran evidencias del circuito

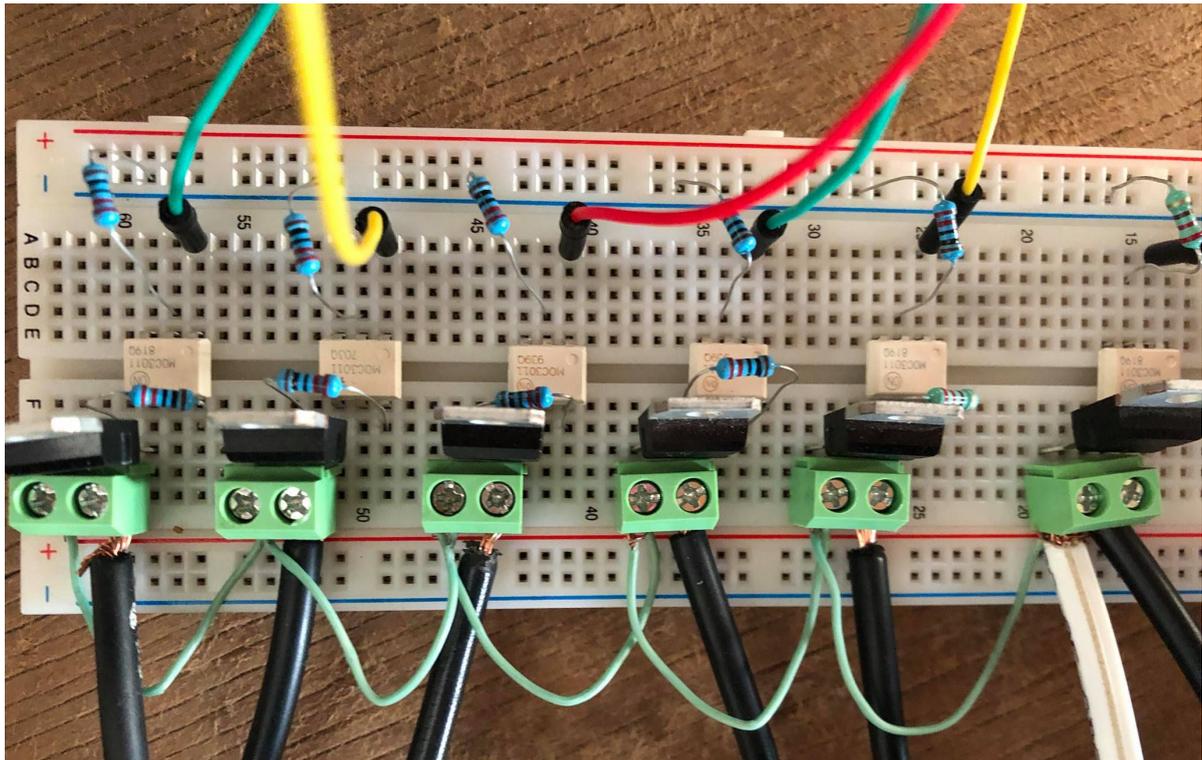


Figura 3. Circuito en el protoboard.

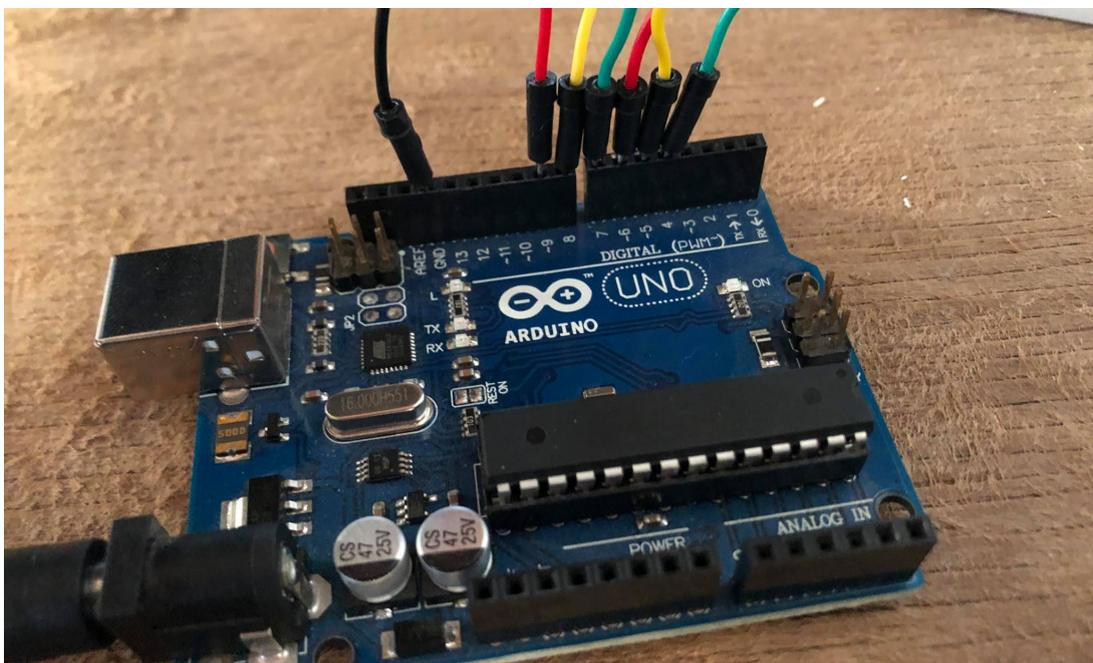


Figura 4. Salidas utilizadas en el arduino, y el GND digital.



Figura 5. Vista general del circuito.

Evidencias del código programado.

Aquí primero le copiare el código que use para mejor lectura:

```
//Aquí declare los pines como cada led, D=Denso y M = Moderado
int GD = 4;
int YD = 5;
int RD = 6;
int GM = 7;
int YM = 8;
int RM = 9;

//Estos son para usar en los tiempos de retardo
int verdeD = 15000; //El led verde de la calle densa dura 15 seg prendido
int verdeM = 10000; //El led verde de la calle moderada dura 10 seg prendido
int ambar = 3000; //El led ámbar de las dos calles duran 3 seg prendido

void setup()
{
  //Se declaran los pines 5-10 de cada led como salida
  pinMode (GM, OUTPUT);
  pinMode (YM, OUTPUT);
  pinMode (RM, OUTPUT);
  pinMode (GD, OUTPUT);
  pinMode (YD, OUTPUT);
  pinMode (RD, OUTPUT);

  digitalWrite(GD,LOW);
  digitalWrite(YD,LOW);
  digitalWrite(RD,LOW);
  digitalWrite(GM,LOW);
  digitalWrite(YM,LOW);
  digitalWrite(RM,LOW);
}

void loop()
{
  //Las banderas seran necesarias para el if que puse, son para los leds rojos
  //Como el ciclo termina cuando el verde moderado se y el ambar moderado se
  //apagan el rojo debera prenderse y se le da ese valor a las banderas de inicializacion
  int bandera1=0;
  int bandera2=0;

  if(bandera2==0)//si la bandera es 0 el led rojoD se apaga y el rojoM se prende
  {
    digitalWrite(RD,LOW);
```

```

digitalWrite(RM,HIGH);
}

digitalWrite(GD,HIGH);//Comienza encendiendoce el led verde de la calle densa
delay(verdeD);//pasa el delay de 15 seg
digitalWrite(GD,LOW);//y luego comienza a parpadear 1 vez
delay(500);
digitalWrite(GD,HIGH);//2da vez
delay(500);
digitalWrite(GD,LOW);
delay(500);
digitalWrite(GD,HIGH);//3ra vez
delay(500);
digitalWrite(GD,LOW);
delay(500);
bandera1=1;//se le da el valor para que pueda entrar el ciclo if a continuacion
digitalWrite(YD,HIGH);//El led ambar se prende
delay(ambar);//Dura un tiempo de 3 seg
digitalWrite(YD,LOW);//y se apaga

if(bandera1==1)//aqui cuando se le dio el valor a la bandera 1 ya pudo entrar
{
digitalWrite(RD,HIGH);//lo que hace que se apague el led denso rojo se prenda y se
apague el rojo moderado
digitalWrite(RM,LOW);
}

//aqui pasa exactamente lo mismo del inicio

digitalWrite(GM,HIGH);//primero se prende el verde moderado
delay(verdeM);//dura un delay de 10 seg
digitalWrite(GM,LOW);//y comienza a parpadear 3 veces
delay(500);
digitalWrite(GM,HIGH);//1ra vez
delay(500);
digitalWrite(GM,LOW);
delay(500);
digitalWrite(GM,HIGH);//2da vez
delay(500);
digitalWrite(GM,LOW);
delay(500);
digitalWrite(YM,HIGH);//3ra vez
delay(ambar);
digitalWrite(YM,LOW);

}

```

Ahora unas evidencias:

```
SEMAFOROLEDS$
//Aquí declare los pines como cada led, D=Denso y M = Moderado
int GD = 4;
int YD = 5;
int RD = 6;
int GM = 7;
int YM = 8;
int RM = 9;

//Estos son para usar en los tiempos de retardo
int verdeD = 15000; //El led verde de la calle densa dura 15 seg prendido
int verdeM = 10000; //El led verde de la calle moderada dura 10 seg prendido
int ambar = 3000; //El led ambar de las dos calles duran 3 seg prendido

void setup()
{
  //Se declaran los pines 5-10 de cada led como salida
  pinMode (GM, OUTPUT);
  pinMode (YM, OUTPUT);
  pinMode (RM, OUTPUT);
  pinMode (GD, OUTPUT);
  pinMode (YD, OUTPUT);
  pinMode (RD, OUTPUT);

  digitalWrite (GD,LOW);
  digitalWrite (YD,LOW);
  digitalWrite (RD,LOW);
  .....
}
```

```
SEMAFOROLEDS$
banderal=1; //se le da el valor para que pueda entrar el ciclo if a continuacion
digitalWrite(YD,HIGH); //El led ambar se prende
delay(ambar); //Dura un tiempo de 3 seg
digitalWrite(YD,LOW); //y se apaga

if(banderal==1) //aquí cuando se le dio el valor a la bandera 1 ya pudo entrar
{
  digitalWrite(RD,HIGH); //lo que hace que se apague el led denso rojo se prenda y se apague el rojo moderado
  digitalWrite(RM,LOW);
}

//aquí pasa exactamente lo mismo del inicio

digitalWrite(GM,HIGH); //primero se prende el verde moderado
delay(verdeM); //dura un delay de 10 seg
digitalWrite(GM,LOW); //y comienza a parpadear 3 veces
delay(500);
digitalWrite(GM,HIGH); //1ra vez
delay(500);
digitalWrite(GM,LOW);
delay(500);
digitalWrite(GM,HIGH); //2da vez
delay(500);
digitalWrite(GM,LOW);
delay(500);
digitalWrite(YM,HIGH); //3ra vez
delay(ambar);
digitalWrite(YM,LOW);
```

Figura 6. Evidencias del código de programación en arduino IDE.

Evidencia de las Simulaciones

Esta la hice en PROTEUS 8 como mencione al inicio, primero se realizó la simulación para ver si no había errores de diseño y una vez comprobado todo se armó en el protoboard.

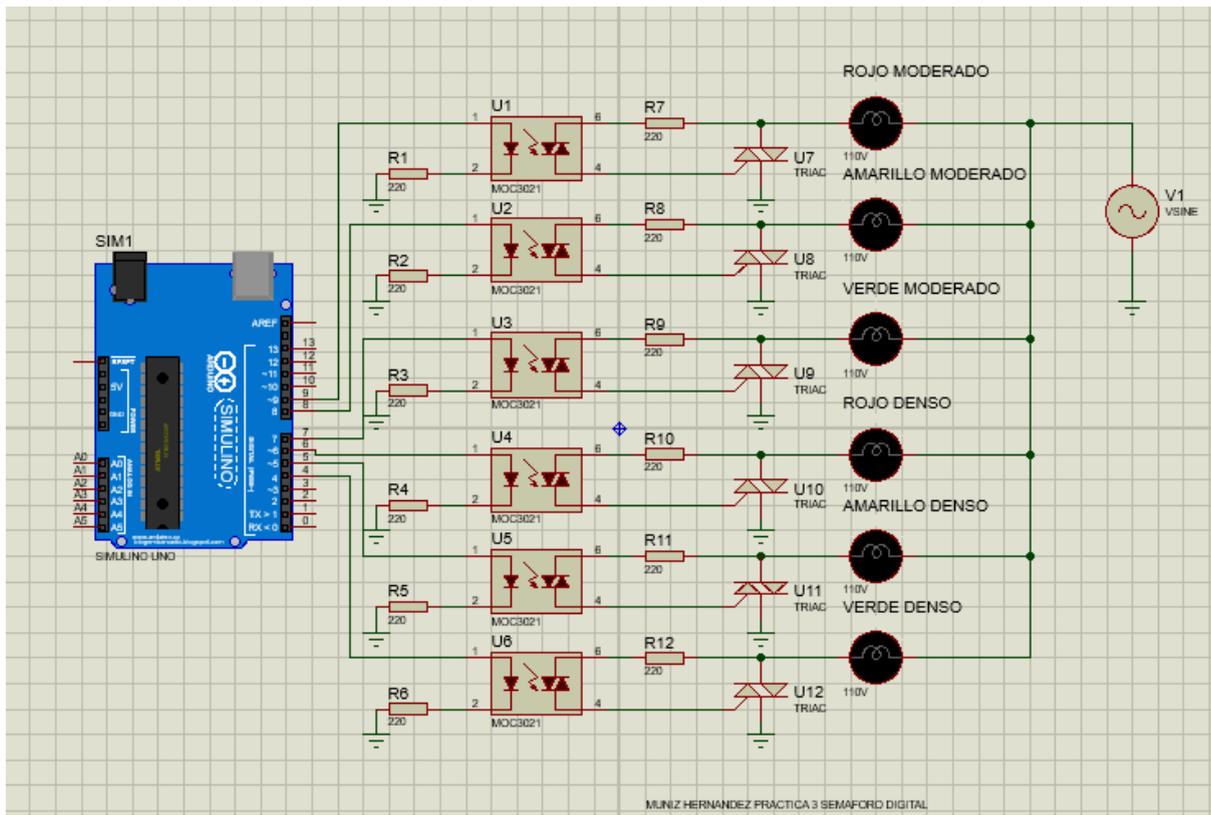


Figura 7. Evidencias del circuito realizado en PROTEUS.

También se le agregó un video de comprobación adjuntado en drive, el cual muestra el comportamiento del circuito.

Conclusiones

Los objetivos de la práctica se cumplieron de manera exitosa, aunque al inicio se me complicó un poco la práctica por el motivo de que no podía conectar el cable duplex al protoboard, si se me alargó mucho el proceso por eso incluso quemé 3 TRIACS por conectarlo de mal forma, también como no sabía como conectar el duplex al protoboard utilice unos jumpers los cuales se quemaron. Preguntando a mis compañeros pude solucionar el problema pero si tuve que comprar los 3 triacs quemados, pero de los errores se aprende y se que no volveré a cometer los mismos errores.

Fuera de eso la práctica fue genial, programar el ciclo de los focos fue sencillo al igual que armar el circuito en físico y en el simulador, solo se debe seguir el diagrama de la figura 1.

Bibliografía

Arduino - What is Arduino & IDE? . Recuperado 30 de agosto de 2021, de <https://www.arduino.cc/en/Guide/Introduction>