



Tecnológico Nacional de México
Instituto Tecnológico de Tijuana
Su dirección Académica
Departamento de Sistemas y Computación
Enero – junio 2017

Serie SC8A

Materia:

Sistemas Programables

Unidad 2

Título:

Práctica 2

Maestro:

Mitre Padilla Luis Alberto

Alumno:

Álvarez Corral Miguel Ángel 13211384

Espinoza Covarrubias Alejandro 13211465

Fecha:

Abril 01 del 2017

Introducción

El uso de los display es muy útil al momento de querer representar de mejor manera valores que se quieran representar, de igual manera se puede implementar el uso de diodos emisores de luz (LED). En esta práctica se requiere con la ayuda de Arduino mostrar los números del 0 a F al igual que con diodos LED mostrar su equivalente binario, así como el uso de un ITR8201 para interrumpir la señal ya que cada vez que se interrumpa la señal los números deberán ir en aumento. En Arduino UNO se deberá cargar el código desarrollado para que este cumpla el funcionamiento requerido.

Marco teórico

Esta práctica será realizada con los siguientes componentes:

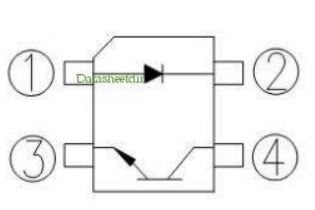
- Protoboard
- Diodo LED
- ITR 8201
- Display de 7 segmentos (Cátodo común)
- Resistencias 220 Ohms
- Arduino UNO

Protoboard: es una especie de tablero con orificios, en la cual se pueden insertar componentes electrónicos y cables para armar circuitos. Como su nombre lo indica, esta tableta sirve para experimentar con circuitos electrónicos, con lo que se asegura un buen funcionamiento del mismo.

Diodo LED: El LED (Light-Emitting Diode: Diodo Emisor de Luz), es un dispositivo semiconductor que emite luz incoherente de espectro reducido cuando se polariza de forma directa la unión PN en la cual circula por él una corriente eléctrica. Su símbolo electrónico es:



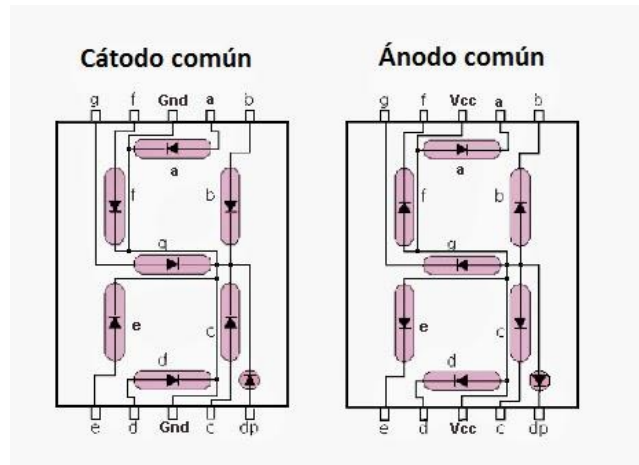
ITR 8102: consiste en un diodo emisor infrarrojo y un Fototransistor NPN de silicio encerrado uno frente a otro convergiendo por medio de un eje óptico. Su representación en diagrama es:



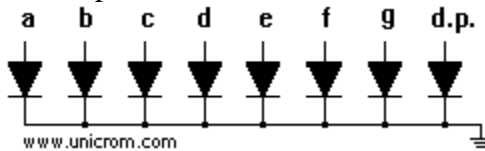
Display de 7 segmentos: El display de 7 segmentos, es un componente que se utiliza para la representación de números en muchos dispositivos electrónicos.

Cada vez es más frecuente encontrar LCD's en estos equipos (debido a su bajísima demanda de energía), todavía hay muchos que utilizan el display de 7 segmentos por su simplicidad.

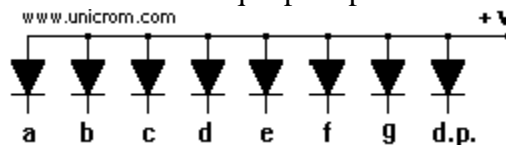
Este elemento se ensambla o arma de manera que se pueda activar cada segmento (diodo LED) por separado logrando de esta manera combinar los elementos y representar todos los números en el display (del 0 al 9). El display de 7 segmentos más común es el de color, por su facilidad de visualización.



1. Display cátodo común: El display cátodo común tiene todos los ánodos de los diodos LED unidos y conectados a tierra. Para activar un segmento de estos hay que poner el ánodo del segmento a encender a Vcc (tensión de la fuente) a través de una resistencia para limitar el paso de la corriente.



2. Display ánodo común: En el display ánodo común, todos los ánodos de los diodos LED unidos y conectados a la fuente de alimentación. En este caso para activar cualquier elemento hay que poner el cátodo del elemento a tierra a través de una resistencia para limitar la corriente que pasa por el elemento.



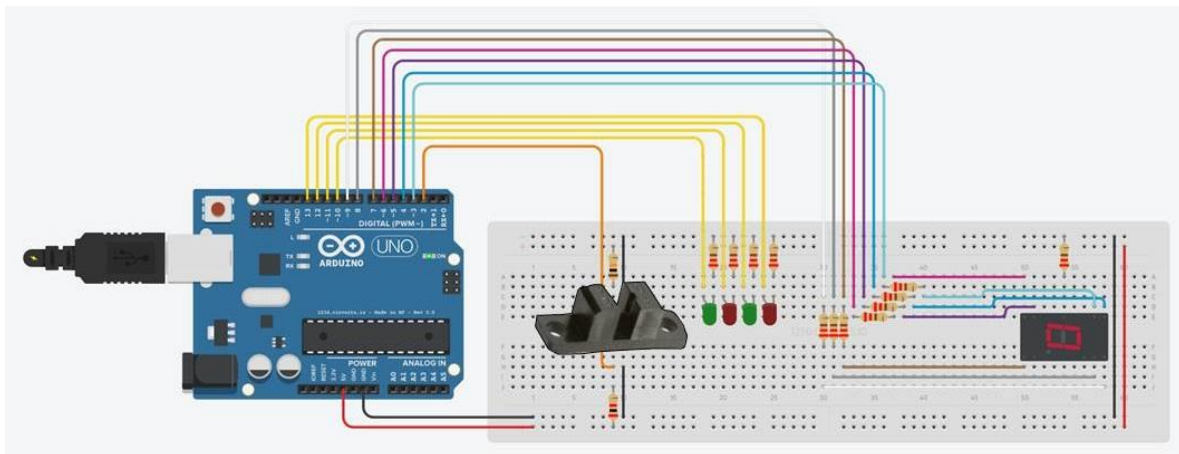
Resistencia: se le denomina resistencia eléctrica a la oposición al flujo de electrones al moverse a través de un conductor. La unidad de resistencia en el Sistema Internacional es el ohmio, que se representa con la letra griega omega (Ω), en honor al físico alemán Georg Simon Ohm, quien descubrió el principio que ahora lleva su nombre. Su símbolo eléctrico es:



Arduino UNO: El Arduino Uno es una placa electrónica basada en el ATmega328 (ficha técnica). Cuenta con 14 pines digitales de entrada / salida (de los cuales 6 se pueden utilizar como salidas PWM), 6 entradas analógicas, un 16 MHz resonador de cerámica, de una conexión USB, un conector de alimentación, una cabecera ICSP, y un botón de reinicio.



Diagrama de Circuit.io



Código de la práctica

// Álvarez Corral Miguel Ángel 13211384

// Espinoza Covarrubias Silverio Alejandro 13211465

// Leds, para número binario

int led1 = 9, led2 = 10, led3 = 11, led4 = 12;

// Declaración de variables para segmentos de display

int a = 8, b = 5, c = 4, d = 3, e = 2, f = 7, g = 6;

//Opto Interruptor

```
int sensor = 13;

// Variable para control de encendido/apagado de leds y display

bool on = true;

//acumulador

int i = 0;

void setup() {

    // Asignación de modo de uso para los pins

    pinMode(led1, OUTPUT);
    pinMode(led2, OUTPUT);
    pinMode(led3, OUTPUT);
    pinMode(led4, OUTPUT);
    pinMode(a, OUTPUT);
    pinMode(b, OUTPUT);
    pinMode(c, OUTPUT);
    pinMode(d, OUTPUT);
    pinMode(e, OUTPUT);
    pinMode(f, OUTPUT);
    pinMode(g, OUTPUT);
    pinMode(sensor, INPUT);
    digitalWrite(led1, !on);
    digitalWrite(led2, !on);
    digitalWrite(led3, !on);
    digitalWrite(led4, !on);
    digitalWrite(a, on);
    digitalWrite(b, on);
```

```

digitalWrite(c, on);
digitalWrite(d, on);
digitalWrite(e, on);
digitalWrite(f, on);
digitalWrite(g, !on);
}

void loop() {
    // Condición que se activa cuando la lectura del sensor es falsa
    if(!digitalRead(sensor)){
        delay(200);
        // Mientras la lectura sea falsa se mantendrá ciclado el programa
        while(!digitalRead(sensor)){ }

        // Una vez que salga del ciclo, si la variable es true, se incrementa el
        acumulador,

        // y se inicia un switch-case para el encendido/apagado de leds y display
        if (on == true)
        {
            i = i+1;
            switch(i)
            {
                // 0
                case 0:
                    digitalWrite(led1, !on);
                    digitalWrite(led2, !on);
                    digitalWrite(led3, !on);
                    digitalWrite(led4, !on);

```

```
    digitalWrite(a, on);
    digitalWrite(b, on);
    digitalWrite(c, on);
    digitalWrite(d, on);
    digitalWrite(e, on);
    digitalWrite(f, on);
    digitalWrite(g, !on);
    break;
// 1
case 1:
    digitalWrite(led1, on);
    digitalWrite(led2, !on);
    digitalWrite(led3, !on);
    digitalWrite(led4, !on);
    digitalWrite(a, !on);
    digitalWrite(b, on);
    digitalWrite(c, on);
    digitalWrite(d, !on);
    digitalWrite(e, !on);
    digitalWrite(f, !on);
    digitalWrite(g, !on);
    break;
// 2
case 2:
    digitalWrite(led1, !on);
```



```
digitalWrite(led2, on);
digitalWrite(led3, !on);
digitalWrite(led4, !on);
digitalWrite(a, on);
digitalWrite(b, on);
digitalWrite(c, !on);
digitalWrite(d, on);
digitalWrite(e, on);
digitalWrite(f, !on);
digitalWrite(g, on);
break;
// 3
case 3:
    digitalWrite(led1, on);
    digitalWrite(led2, on);
    digitalWrite(led3, !on);
    digitalWrite(led4, !on);
    digitalWrite(a, on);
    digitalWrite(b, on);
    digitalWrite(c, on);
    digitalWrite(d, on);
    digitalWrite(e, !on);
    digitalWrite(f, !on);
    digitalWrite(g, on);
    break;
```

```
// 4
```

```
case 4:
```

```
    digitalWrite(led1, !on);  
    digitalWrite(led2, !on);  
    digitalWrite(led3, on);  
    digitalWrite(led4, !on);  
    digitalWrite(a, !on);  
    digitalWrite(b, on);  
    digitalWrite(c, on);  
    digitalWrite(d, !on);  
    digitalWrite(e, !on);  
    digitalWrite(f, on);  
    digitalWrite(g, on);  
    break;
```

```
// 5
```

```
case 5:
```

```
    digitalWrite(led1, on);  
    digitalWrite(led2, !on);  
    digitalWrite(led3, on);  
    digitalWrite(led4, !on);  
    digitalWrite(a, on);  
    digitalWrite(b, !on);  
    digitalWrite(c, on);  
    digitalWrite(d, on);  
    digitalWrite(e, !on);
```

```
    digitalWrite(f, on);
    digitalWrite(g, on);
    break;
// 6
case 6:
    digitalWrite(led1, !on);
    digitalWrite(led2, on);
    digitalWrite(led3, on);
    digitalWrite(led4, !on);
    digitalWrite(a, on);
    digitalWrite(b, !on);
    digitalWrite(c, on);
    digitalWrite(d, on);
    digitalWrite(e, on);
    digitalWrite(f, on);
    digitalWrite(g, on);
    break;
// 7
case 7:
    digitalWrite(led1, on);
    digitalWrite(led2, on);
    digitalWrite(led3, on);
    digitalWrite(led4, !on);
    digitalWrite(a, on);
    digitalWrite(b, on);
```

```
    digitalWrite(c, on);
    digitalWrite(d, !on);
    digitalWrite(e, !on);
    digitalWrite(f, !on);
    digitalWrite(g, !on);
    break;
// 8
case 8:
    digitalWrite(led1, !on);
    digitalWrite(led2, !on);
    digitalWrite(led3, !on);
    digitalWrite(led4, on);
    digitalWrite(a, on);
    digitalWrite(b, on);
    digitalWrite(c, on);
    digitalWrite(d, on);
    digitalWrite(e, on);
    digitalWrite(f, on);
    digitalWrite(g, on);
    break;
// 9
case 9:
    digitalWrite(led1, on);
    digitalWrite(led2, !on);
    digitalWrite(led3, !on);
```

```
digitalWrite(led4, on);
digitalWrite(a, on);
digitalWrite(b, on);
digitalWrite(c, on);
digitalWrite(d, on);
digitalWrite(e, !on);
digitalWrite(f, on);
digitalWrite(g, on);
break;
// A
case 10:
    digitalWrite(led1, !on);
    digitalWrite(led2, on);
    digitalWrite(led3, !on);
    digitalWrite(led4, on);
    digitalWrite(a, on);
    digitalWrite(b, on);
    digitalWrite(c, on);
    digitalWrite(d, !on);
    digitalWrite(e, on);
    digitalWrite(f, on);
    digitalWrite(g, on);
    break;
// B
case 11:
```

```
digitalWrite(led1, on);
digitalWrite(led2, on);
digitalWrite(led3, !on);
digitalWrite(led4, on);
digitalWrite(a, !on);
digitalWrite(b, !on);
digitalWrite(c, on);
digitalWrite(d, on);
digitalWrite(e, on);
digitalWrite(f, on);
digitalWrite(g, on);
break;
// C
case 12:
    digitalWrite(led1, !on);
    digitalWrite(led2, !on);
    digitalWrite(led3, on);
    digitalWrite(led4, on);
    digitalWrite(a, !on);
    digitalWrite(b, !on);
    digitalWrite(c, !on);
    digitalWrite(d, on);
    digitalWrite(e, on);
    digitalWrite(f, !on);
    digitalWrite(g, on);
```

```
    break;
// D
case 13:
    digitalWrite(led1, on);
    digitalWrite(led2, !on);
    digitalWrite(led3, on);
    digitalWrite(led4, on);
    digitalWrite(a, !on);
    digitalWrite(b, on);
    digitalWrite(c, on);
    digitalWrite(d, on);
    digitalWrite(e, on);
    digitalWrite(f, !on);
    digitalWrite(g, on);
    break;
// E
case 14:
    digitalWrite(led1, !on);
    digitalWrite(led2, on);
    digitalWrite(led3, on);
    digitalWrite(led4, on);
    digitalWrite(a, on);
    digitalWrite(b, !on);
    digitalWrite(c, !on);
    digitalWrite(d, on);
```

```

    digitalWrite(e, on);
    digitalWrite(f, on);
    digitalWrite(g, on);
    break;
// F
default:
    digitalWrite(led1, on);
    digitalWrite(led2, on);
    digitalWrite(led3, on);
    digitalWrite(led4, on);
    digitalWrite(a, on);
    digitalWrite(b, !on);
    digitalWrite(c, !on);
    digitalWrite(d, !on);
    digitalWrite(e, on);
    digitalWrite(f, on);
    digitalWrite(g, on);

    // El acumulador se pone en -1 para que a la siguiente activación
    despliegue 0
    i = -1;
    break;
}
}
}
}

//===== FIN DE CODIGO =====

```


Diagrama real



Conclusión

Utilizando ArduinoUNO y un protoboard, se comprobó el funcionamiento de los display el resulta muy al momento de querer mostrar valores ya de manera más visual y utilizar el ITR 8201 para el incremento de los valores, es importante conocer los componentes que tienen polaridad y los que no, ya que esto es un factor importante al momento de conectar los componentes. También se desarrolló un código el cual sería necesario para poder realizar el funcionamiento adecuado de los componentes antes mencionados. En esta práctica se pudo apreciar también las distintas maneras en la que se pueden utilizar los LED ya que se representaron los números de forma binaria lo cual puede ser un uso muy útil cuando no se cuenta con un display.