

Práctica No. 2 del Curso "Microcontroladores"

Uso de las entradas salida digitales multiplexadas

Objetivos

Usar el Arduino para manejar más salidas digitales mediante la multiplexación.

Material

- Resistencias de 100Ω o 220Ω .
- 27 LED's
- 3 transistores BC547 o similar
- 3 resistencia de $1k\Omega$.
- Arduino

Planteamiento

Cuando se necesitan más salidas que las que físicamente posee un dispositivo se puede usar la técnica de multixación. La cual funciona para indicadores visules, debido a la baja velocidad de respuesta que tiene el ojo. Si la frecuencia de conmutación de los LED's es mayor a 60Hz, de un efecto de continuidad, como si todo el tiempo estuviera prendida.

En esta práctica se van a controlar 27 LED's con 11 salidas de la placa de Arduino. Para lo cual se va a armar un cubo de LED's como el que se muestra en la figura 1. El cual consta de 3 pisos y en cada uno de ellos se tiene 9 LED's. Se van a utilizar 3 salidas para seleccionar el piso a encender y 9 salidas para direccionar cada uno de los nueve LED's de cada piso.

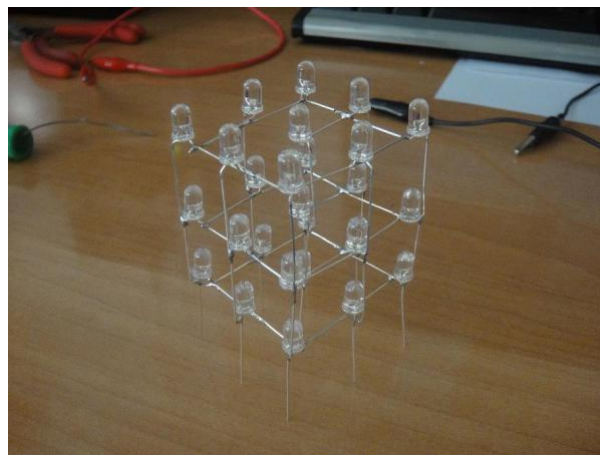


Figura 1. Cubo de LED's de 3x3x3

En cada piso los cátodos de todos los 9 LED's se interconectan y los ánodos de cada LED que este en la misma fila se conectan. Estos últimos van a ser controlados por cada una de las 9 salidas digitales. En la figura 2 se muestra la forma en que están interconectados para un piso. Para armarlo es necesario hacer una plantilla en unicel con el fin de poder colocarlos con la cabeza hacia abajo para evitar que se muevan y que queden equidistantes. Una vez colocado se les aplica un punto de soldadura para unirlos.

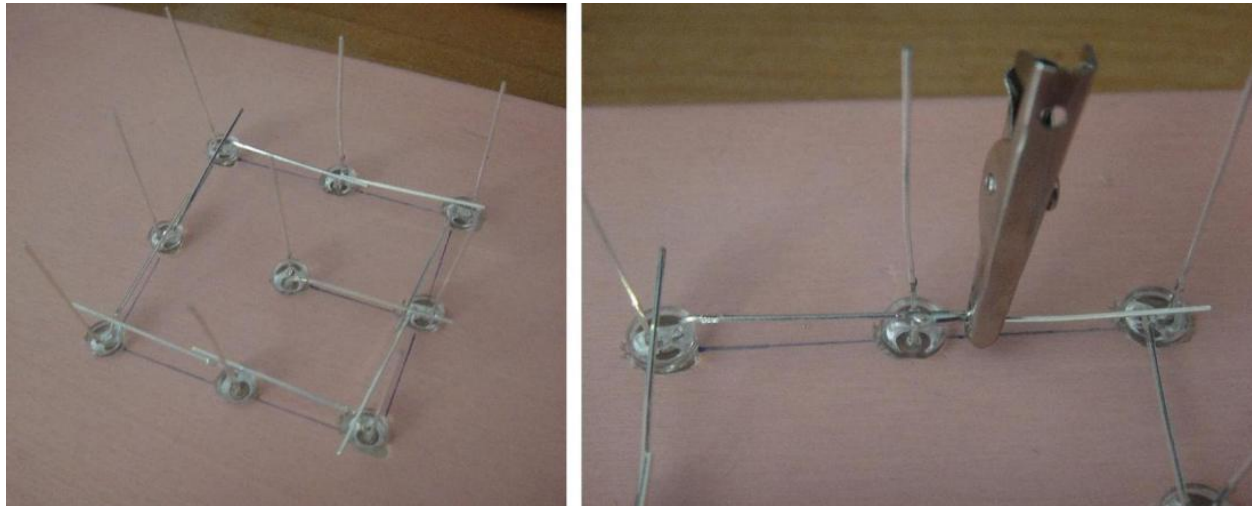


Figura 2. Piso del Cubo con 9 LED's

Procedimiento.

En la figura 3 se muestra el diagrama de conexiones del cubo de LED's y el Arduino. Las columnas se controlan con las salidas digitales 1 a la 9, y cada fila se conecta con las salidas 10 a la 12. Para aumentar la capacidad de corriente, estas tres salidas controlan a 3 transistores tipo NPN (BC547), que permite manejar toda la corriente de los LED's de cada piso. Las resistencias de cada LED son de 100Ω o 220Ω para limitar la corriente. La resistencia de base de los transistores es de 1kΩ. En la tabla 1 se muestra el código de ejemplo para encender los LED's

Tabla 1. Programa de prueba del cubo de LED's

```
int Columnas[] = {1, 2, 3, 4, 5, 6, 7, 8, 9};

int Filas[] = {12, 11, 10};
int RandFila;      //Variable para fila aleatoria
int RandColumna;  //Variable para columna aleatoria

void setup()
{
    int contador;

    for (int contador = 1; contador < 10; contador++){
        pinMode(Columnas[contador], OUTPUT); }

    for (int contador = 1; contador < 4; contador++){
        pinMode(Filas[contador], OUTPUT); }
}

void loop()
{
    RandLed();
}

void RandLed()
{
    RandFila = random(0,3);
```

```
RandColumna = random(0,9);  
  
digitalWrite(Filas[RandFila], HIGH);  
digitalWrite(Columnas[RandColumna], HIGH);  
  
delay(75);  
  
digitalWrite(Filas[RandFila], LOW);  
digitalWrite(Columnas[RandColumna], LOW);  
  
delay(50);  
}
```

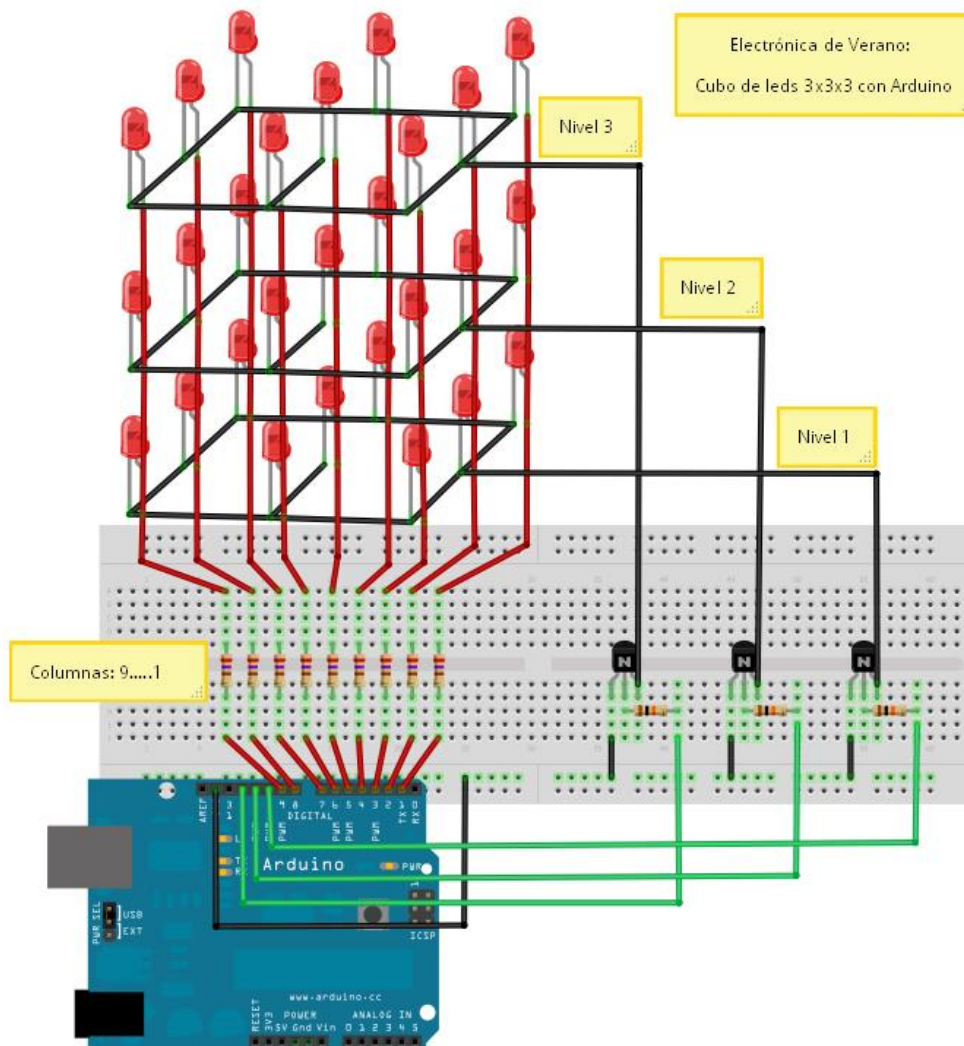


Figura 3. Diagrama de conexiones del cubo de LED's con Arduino

Hacer un programa en el cual encienda todos los LED's piso por piso en forma de barrido. Posteriormente hacer el programa que haga un barrido de columna por columna, que encienda todos los LED's del cubo y los de las orillas.

Reporte

El reporte se va a entregar en 1 semanas después de concluida la práctica. Se va utilizar el formato de la IEEE. No se les olvide incluir, la introducción, el procedimiento, resultado, conclusiones y bibliografía.

Referencias.

- [URL1] Cubo de LED's. <http://electronicavm.net/2011/07/27/cubo-de-leds-3x3x3-con-arduino/>
 . Fecha del último acceso 30/octubre/2015