



Actividad	Inciso	Nombre
- Propuesta Proyecto Final	-	Matriz LED

Introducción

Como parte del curso, se ha propuesto el hacer un programa que controle algo físico, hacer una interfaz e interactuar con ella.

La decisión para esta actividad, ha sido construir una matriz LED de 8 x 8, la cual *dibujará* lo que le sea indicado en el programa de computadora, ya sea una letra sencilla, un patrón, o una demostración del control único de cada LED.

Material Utilizado

Esta lista comprende los elementos empleados tanto en software como en hardware que se emplearán para la actividad

Software

- Sistema Operativo basado en Linux (Kernel 4.4.3.1) (Distribución Específica: Arch Linux x64)
- NASM (NetWide Assembler) v2.11.8
- GNU Linker (ld) v3.7
- GNU Debugger (GDB) v7.11-1
- KDE Debugger (KDGB) v2.5.5-3
- Editor de texto **Sublime Text** (no esencial para realizar el proceso de ensamble/ligado)
-

Hardware

Computadora

- Intel i5-3210M

Construcción física del circuito

- Arduino MEGA 2560
- Tarjeta Perforada
- 64 LED's blancos
- Registro de Corrimiento (shift register) de 8 bits **74HC595**
- Arreglo Darlington **ULN2803**

Descripción Básica del Proyecto

La matriz se compondrá de la siguiente manera:

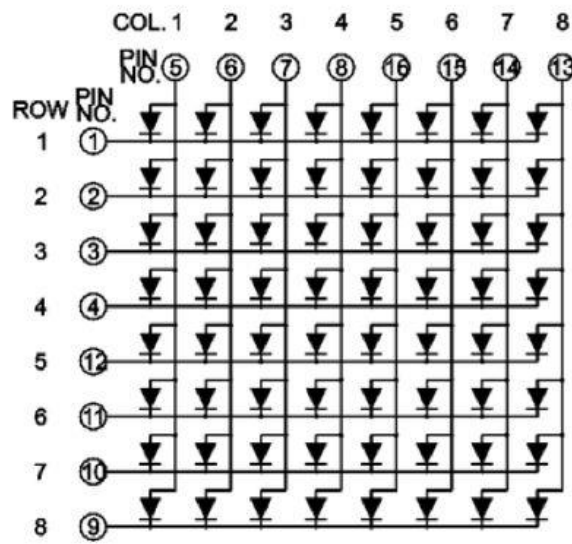


Fig 1. Diagrama de conexión de la matriz de 8x8

Esta matriz será controlada (en energía) por parte del registro **74HC595**, el cual actuará como *memoria* de video:

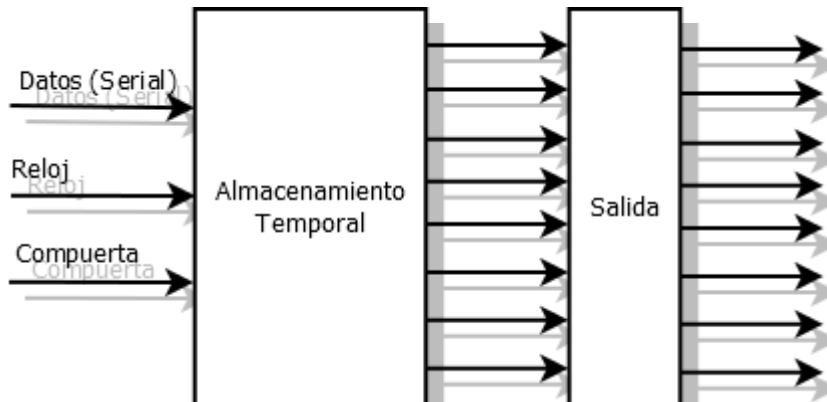


Fig 2. Diagrama de bloques del IC 74HC595

El circuito integrado recibe por medio de una comunicación serial, 1 byte de información, el cual es guardado en el almacenamiento temporal, después, cuando se activa la compuerta, hace una salida en paralelo a la sección de salida. Cabe mencionar que el almacenamiento y la salida son **2 memorias distintas**, es decir, técnicamente el registro consta de **2 bytes** de almacenamiento.

Este integrado lo que hará será activar los LED's necesarios de acuerdo al byte que se le envíe, por ejemplo:

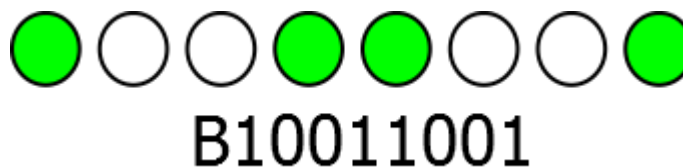


Fig 3. Demostración de salida del registro

El registro recibe ese byte de manera serial, después, lo saca de manera paralela, activando cada led correspondiente. De esta manera, se tiene un mapeo **1 a 1** de la *memoria de video* y el *display*.

El truco, como en cualquier cosa, yace en algo que se llama **scanning**, el cual, por la misma construcción de la matriz, **no nos dejaría encender** todos los LED's de manera simultánea, pero gracias al scanning y a la **persistencia de visión**, podemos simular esto.

Aquí es donde toma parte el arreglo **ULN2803**, este lo que hará será poner a tierra toda la fila que se le indique, esto, combinado con un alto radio de **refresco**, hace parecer que la matriz está encendida de una sola vez, cuando en realidad se enciende una fila, se apaga y se pasa a la siguiente.

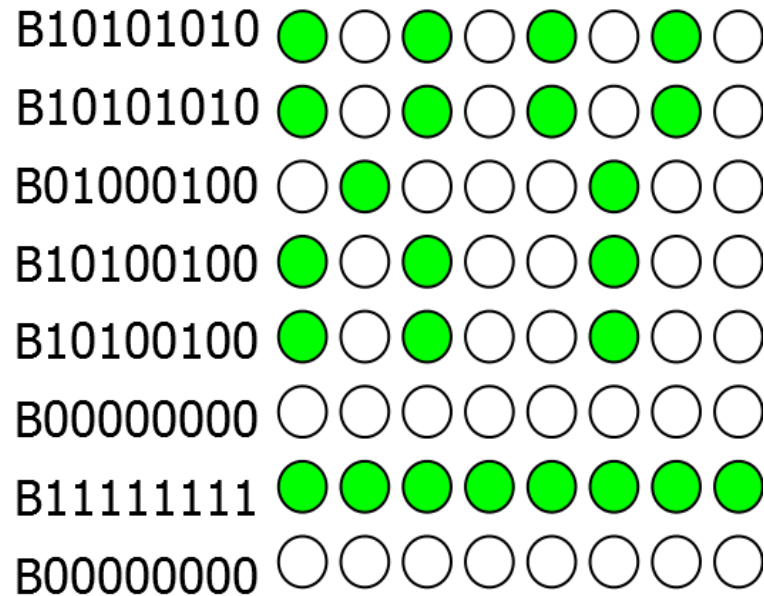


Fig 4. Demostración de la matriz con un patrón.

Así pues, en total se estarían enviando un total de **64 bits** de información, los cuales representarían la cantidad total de LED's disponibles.

Teoría del programa

El programa de PC en ensamblador abarcará los siguientes puntos:

- Comunicación serial con el Arduino
- Manejo de la **memoria de video**
- Construcción de patrones y secuencias
- Control de tiempos

El Arduino sólo constará de una rutina simple, la cual abarca lo siguiente:

- Esperar información serial
- Procesar información y enviarla al registro
- Realizar la alternación de las filas a una velocidad aproximadamente constante

Referencias

El proyecto de una matriz no es algo nuevo, y se toma referencia de los siguientes proyectos para poder completar este:

1. <http://embedded-lab.com/blog/making-a-8x40-led-matrix-marquee-using-shift-registers/>
2. <http://www.instructables.com/id/LED-matrix-using-shift-registers/>
3. <http://www.instructables.com/id/Beginner-tutorial-Controlling-LED-matrix-with-2-59/>