

Parcial 1

Informe

Jhonny Alejandro Ortiz Osorio

C.C: 1001015092

Juan José Florez Argáez

C.C: 1001765286

Departamento de Ingeniería Electrónica y

Telecomunicaciones

Universidad de Antioquia

Medellín

Marzo de 2021

Índice

1. Sección introductoria	2
2. Sección de contenido	2
2.1. Análisis del problema	2
2.2. Esquema de trabajo	2
2.3. Pruebas de desarrollo	2
2.4. Problemas encontrados	3
2.5. Evolución del algoritmo	3
2.6. Algoritmo implementado	5
3. Inclusión de imágenes	5

1. Sección introductoria

Pondremos a prueba nuestras destrezas en análisis de problemas, manejo del lenguaje C++ y la plataforma arduino a través de un gran reto que integra todos los conocimientos enseñados hasta la fecha, el cual tendrá unos requisitos que harán aun mas interesante el desafío.

2. Sección de contenido

Esta sección es para agregar toda la información correspondiente con código, citas, etc.

2.1. Análisis del problema

Principalmente debemos entender el funcionamiento del integrado 74HC595. Luego el diseño y planificación de todas las conexiones necesarias para lograr un circuito estable, eficiente y por ultimo el diseño y desarrollo del software que controlará nuestra matriz de leds.

2.2. Esquema de trabajo

- Estudiar el integrado 74HC595.
- Diseño del circuito.
- Diseño inicial (Idea) del software.
- Diseño de las funciones necesarias.
- Pruebas del software.
- Diseño final del circuito.
- Redacción del manual de usuario.

2.3. Pruebas de desarrollo

- Se probó que todas las conexiones del circuito estén correctas.
- Se realizaron pruebas individuales a cada función del programa.
- Se probaron múltiples patrones y diseños en los leds

2.4. Problemas encontrados

- El primer problema fue averiguar como el usuario va a elegir el o los patrones mostrados en la matriz de leds de forma fácil e intuitiva.
- Creación de la interfaz en forma de teclado para que cada posición del teclado se asociara a un led.
- Un problema muy particular fue con la plataforma tinkercad ya que ésta fallaba muy a menudo.
- Y el problema mas grande fue el manejo de la memoria dinámica.
- Cuando se realizaron las conexiones se nos quemaron dos integrados debido a que uno de ellos estaba al revés.
-

2.5. Evolución del algoritmo

1. Función verificación: Esta función permite verificar la funcionalidad de todos los LED's.
 - Declaramos una matriz de tipo short int de [8][8] con el nombre matriz LED la cual servirá para determinar el estado de los LED's siendo 1 alto y 0 bajo.
 - Implementamos un ciclo for que sirve para encender o mantener apagados los LED's dependiendo del valor que tengan.
 - Luego otro ciclo for que enciende todos los leds a la vez.
2. Función Imagen. Esta función permite que el usuario dibuje su propio patrón y después pueda visualizar el patrón en la matriz LED
 - Declaramos una matriz de tipo short int de [8][8] con el nombre panel LED
 - Definimos la función patrones de tipo char de [8][8].
 - Invocamos las funciones Muestra posiciones y Realizando Dibujo(patrones) esta última con el argumento patrones.
 - La matriz patrones junto con la función Realizando Dibujo muestra en el monitor serial una matriz de 8x8 con los espacios en blanco y va modificando los espacios con una 'X' donde el usuario desee.
 - Se implementa un ciclo for para determinar el estado de los LED's dependiendo de las elecciones del usuario.
 - Luego otro ciclo for que enciende los LED's dependiendo del estado en la matriz panel LED
3. Función Limpieza Pantalla: Esta función limpia la pantalla

- Con un ciclo for se limpia la pantalla que ve el usuario.
4. Función Muestra posiciones: Esta función le muestra al usuario una matriz 8x8 donde cada posición va a tener un valor que representa un led.
 - Definimos la matriz patrones[8][8] de tipo int la cual representa en pantalla los valores que se le asignaran a cada LED.
 - Con un ciclo for anidado dentro de otro ciclo for y varios condicionales se imprime en el monitor serial una matriz 8x8 donde cada espacio va a tener un valor asignado
 5. Función Muestra Patrón Dibujo: Esta función le muestra al usuario una matriz 8x8 donde cada posición va a estar vacía o con una 'X' dependiendo del estado de la matriz dibujo.
 - Se define la matriz dibujo[8][8] de tipo char
 - Luego con dos ciclos for uno anidado dentro de otro y dos condicionales se dibuja una 'X'.
 6. Función Realizando Dibujo: Esta función le muestra al usuario una matriz 8x8 donde cada posición va a estar vacía y puede ser utilizada para dibujar el patrón que se desee.
 - La función recibe como argumento la matriz patrones [8][8]
 - Luego se invoca la función Muestra Patrón Dibujo(patrones) con el argumento patrones para mostrar el teclado con el cual va a poder elegir el led que dese encender.
 - Con un ciclo while se le pide al usuario que ingrese la posición que desee de la matriz 8x8, teniendo en cuenta que la posición que elija significa que ese LED va a ser encendido.
 - Luego con otro ciclo while y varios condicionales se va modificando la matriz que muestra el patrón dibujado.
 7. Función Pubik: Esta función mostrará una secuencia de patrones elegidos por el usuario en un tiempo también elegido por el usuario.
 - Se le pedirá al usuario que ingrese: Número de patrones a mostrar, tiempo que desea ver el patrón en segundos.
 - Se reserva espacio en el Heap dependiendo de la cantidad de patrones que desee el usuario.
 - En el ciclo anterior se van creando los patrones que desea mostrar el usuario con ayuda de la función Realizando Dibujo y además se van agregando los patrones en orden al espacio de memoria en el Heap(se agregan los patrones como 1 y 0 siendo 1 un High y 0 un Low).
 - Con ayuda de otro ciclo se encienden los patrones que puso el usuario y que hasta el momento están almacenados en el Heap.
 - Se libera el espacio reservado en el Heap.

2.6. Algoritmo implementado

3. Inclusión de imágenes

A continuación se presentan imágenes de los componentes utilizados y el proceso en general.

1. En la Figura (1), se presenta el integrado 74HC595.

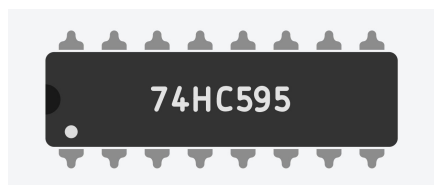


Figura 1: integrado 74HC595

2. En la Figura (2), se presenta el Arduino UNO R3.

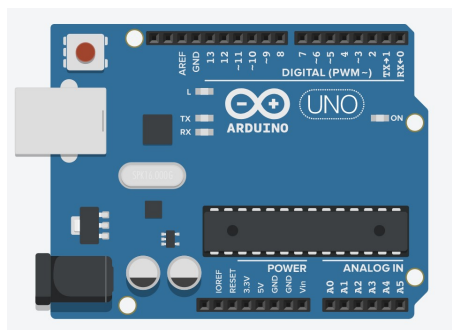


Figura 2: ARDUINO UNO R3

3. En la Figura (3), se presenta el LED.
4. En la Figura (4), Se presenta la placa de pruebas.
5. En la Figura (5), se presenta la resistencia.

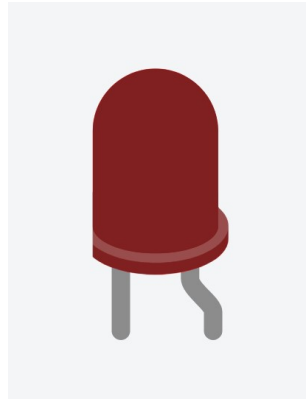


Figura 3: LED

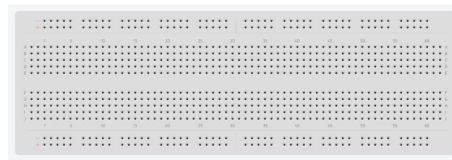


Figura 4: Placa de pruebas grande.

6. En la Figura (6), se presenta la conexión de los objetos a la potencia y la tierra.
7. En la Figura (7), representa la conexión de los pines de Arduino a la entrada y salida de reloj de los integrados.

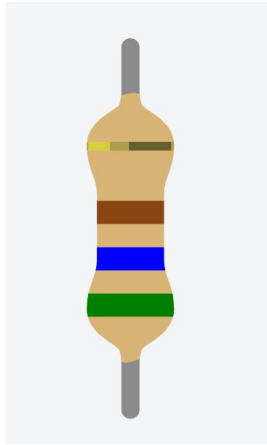


Figura 5: Resistencia

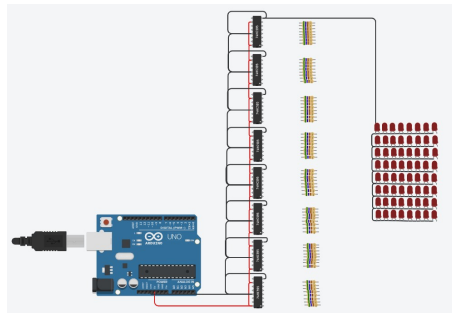


Figura 6: Conexión de los objetos a la potencia y la tierra

8. En la Figura (9), representa la conexión de los pines de Arduino a la entrada y salida de datos de los integrados.
9. En la Figura (??), Conexiones de las resistencias.

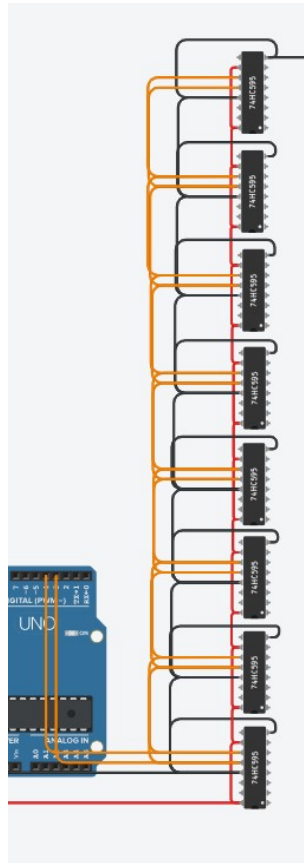


Figura 7: Conexión de los pines de reloj de los integrados

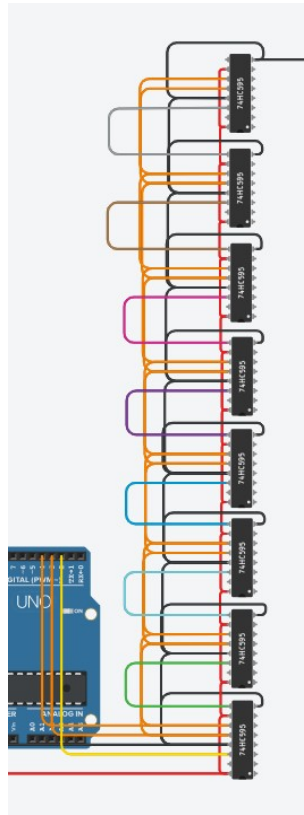


Figura 8: Conexión de entrada y salida de los integrados

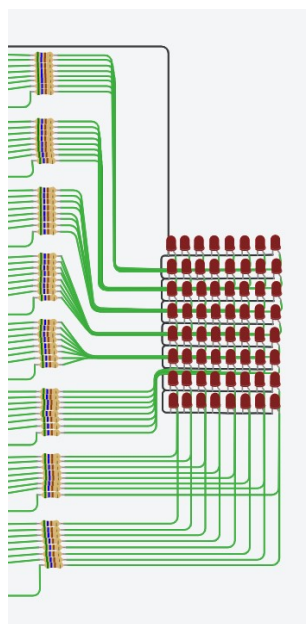


Figura 9: Conexión de las resistencias a los leds.