

Parcial 1 Informática II:
Análisis y diseño matriz de LEDs con Tinkercad y C++

Yuliana Betancur Saldarriaga
Andres Felipe Sepulveda Rivillas

Universidad de Antioquia
Facultad de ingeniería

Informática II
Medellín – Antioquia

En el presente documento se encuentra detallado el proceso de análisis y diseño para la implementación de un sistema de control para una matriz 8x8 compuesta de LEDs, haciendo uso de del entorno de simulación de Tinkercad y el lenguaje de programación C++.

El objetivo de esta implementación es realizar el control de salidas digitales de un microcontrolador para lograr visualizar diferentes patrones por medio de una matriz estructurada por 64 LEDs.

Análisis de requerimientos

1. El sistema debe contar con la capacidad de controlar 64 LEDs organizados en una matriz de 8x8.
2. Los LEDs utilizados deben ser encendidos y apagados de forma individual.
3. En la definición para el uso de los LEDs debe permitirse la visualización de diferentes patrones de luz en la matriz.
4. Debe ser posible ajustar los tiempos de funcionamiento de los diferentes patrones.
5. Las herramientas Tinkercad y C++ deben ser utilizadas como medio para desarrollar la implementación.
6. Uso controlado de recursos de manera eficiente, entre estos un máximo de 7 pines conectados en Arduino.
7. El usuario final debe contar con la capacidad de ingresar comandos por el monitor serial.

Integrado 74HC595

El integrado es bastante útil en este caso para reducir la cantidad de pines de salida utilizados en Arduino de manera eficiente.

Puede ser utilizado para la conexión de salidas a las columnas y filas, cada salida puede controlar cada columna o cada fila de la matriz, además, se puede hacer uso del registro de desplazamiento para enviar datos en serie que representen el patrón deseado en cada columna de la matriz.

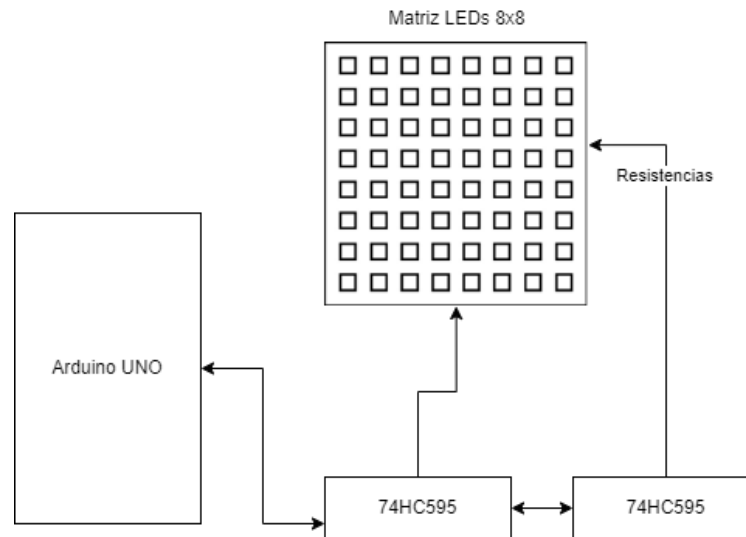
Materiales contemplados

- Microcontrolador (Arduino Uno).
- 64 LEDs para construcción de matriz.
- 8 Resistencias para cada fila de LEDs.
- Cables de conexión.
- 2 integrados 74HC595.

Diseño

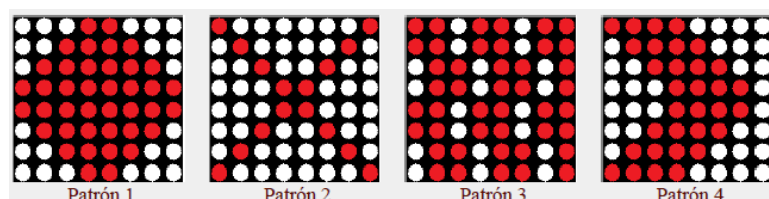
Inicialmente será creada una matriz de 8x8 (8 filas y 8 columnas) constituida por LEDs que estarán conectados entre ellos y a 2 integrados 74HC595, con uno de estos por el lado de las filas y con el otro por el lado de las columnas, así mismo se hará uso de diversas resistencias en serie con cada fila de los LEDs para limitar el paso de corriente.

Uno de los integrados estará conectado al Arduino y posteriormente será conectado al siguiente integrado por las conexiones correspondientes para la distribución de cargas y correcto funcionamiento de ambos para el manejo de los LEDs conectados en la matriz.



Funcionalidades principales

- Inicialización de pines de salida de Arduino.
- Menú para gestión de usuario en el control de luces y manejo de patrones.
- Implementación de funciones para encendido y apagado de LEDs específicos.
 - Función **principal**: Gestión de menú principal.
 - Función **verificación**: Encendido y apagado de los 64 LEDs.
 - Función **imagen**: Ingreso de un patrón definido por el usuario señalando filas y columnas.
 - Función **patrones**: Ejecutar 4 patrones definidos.



- Función **publik**. Permite al usuario seleccionar diferentes opciones por medio de uso de comandos validar el funcionamiento de LEDs, ingresar imagen de prueba y visualizar de forma alternada los patrones del 1 al 4.
- Uso de temporizadores para control de tiempo de encendido y apagado.
- Implementación de ciclos para el manejo del funcionamiento del sistema.

El análisis de la situación y diseño de la implementación fue llevado a cabo teniendo en cuenta los componentes necesarios para poder cumplir con los requisitos planteados en el documento inicial, teniendo en cuenta los elementos que proporcionan una solución viable para la necesidad inicial. La arquitectura propuesta brinda una eficiente distribución de cargas y funcionalidades entre el microcontrolador y los integrados donde puede permitirse la visualización de los patrones de luz y uso de diferentes funciones que serán implementadas en el lenguaje C++.

El paso a seguir incluye realizar la implementación y pruebas correspondientes del sistema en Tinkercad en conjunto con el código implementado y ajustes necesarios de acuerdo a la funcionalidad que tenga el sistema.