

INGENIERÍA MECATRÓNICA



DI_CERO

DIEGO CERVANTES RODRÍGUEZ

ELECTRÓNICA DIGITAL: CIRCUITOS LÓGICOS, LENGUAJE VHDL Y VERILOG

XILINX (64-BIT PROJECT NAVIGATOR) & ADEPT

Matriz Led 8X8

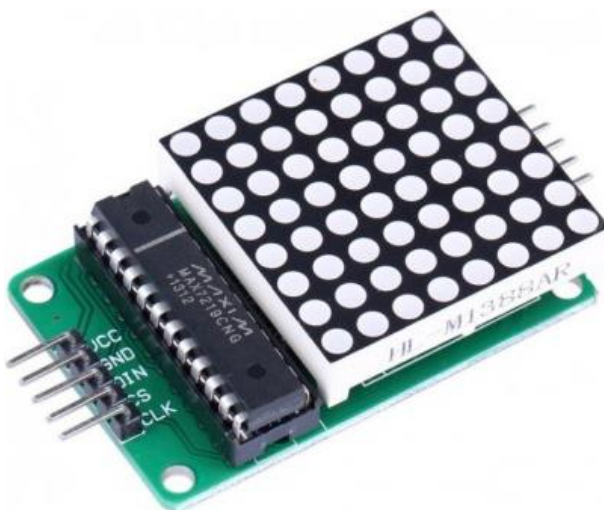
Contenido

Matriz Led 8X8.....	2
Código Arduino Secuencia Encendido y Apagado un Solo Led en Cascada:.....	4
Código Arduino Secuencia Encendido de Todos Leds de la Matriz:	5
Código Arduino Animación de Mostrar Números en la Matriz 8X8:	5
Referencias.....	8



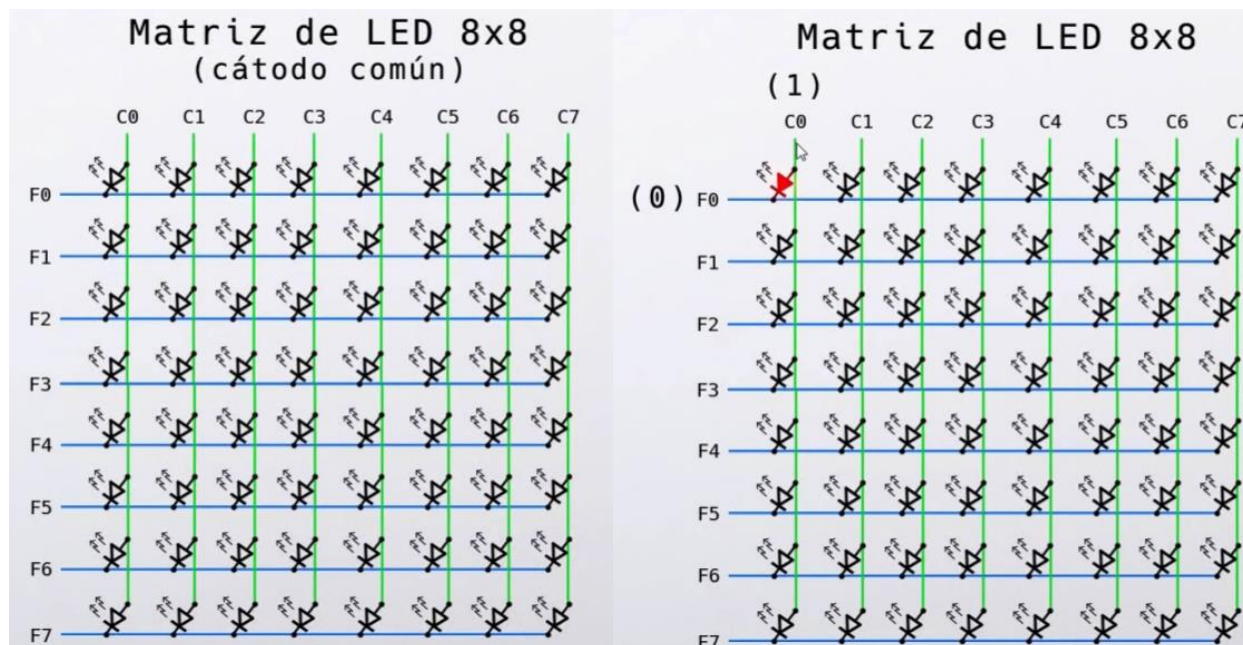
Matriz Led 8X8

El módulo consta de una matriz de 8 leds en sus filas y 8 leds en sus columnas, dando un total de 64 foquitos donde se pueden mostrar letras, números o animaciones, además se cuenta con el **circuito integrado MAX7219** que simplifica el control de la matriz y permite enlazar varias matrices entre sí para mostrar animaciones más largas o grandes.



Los leds del módulo son de **cátodo común**, lo que significa que se activarán cuando se cumplan las siguientes condiciones en un led cuya **fila** se cruza con una **columna** específica:

- Se mande un **0 lógico** a cada una de las **filas** de la matriz de leds 8X8.
- Se mande un **1 lógico** a cada una de las **columnas** de la matriz de leds 8X8.



Para controlar el encendido de leds no será necesario mandar los 16 bits de control, para eso funciona el **circuito integrado MAX7219**, que está diseñado para controlar matrices de leds y displays de 7 segmentos.

Los pines del **MAX7219** funcionan a través de un protocolo de Bus SPI (Serial Peripheral Interface), el cual sirve para poder controlar varias matrices a la vez y sus pines ejecutan las siguientes acciones:

- **ISET:** Pin VCC que ya tiene una resistencia interna incluida y recibe de a **5V de alimentación**.
- **DIN:** Pin Master Out Slave In (**MOSI**) que **indica la dirección de circulación de datos**, la cual viene del microcontrolador, FPGA o CPU y se dirige hacia el circuito integrado MAX7219.
- **LOAD:** Pin Chip Select (**CS**) que **se activa con nivel bajo** (0 lógico) y **habilita la función del esclavo**, osea **el circuito integrado MAX7219**.
- **CLK:** Pin de reloj serial que **indica el pulso de reloj que marca el paso con el que se mandan los datos del maestro al esclavo**.
- **DOUT:** Pin que **permite encadenar múltiples módulos usando el protocolo SPI** para gestionar todos a través de un mismo controlador.

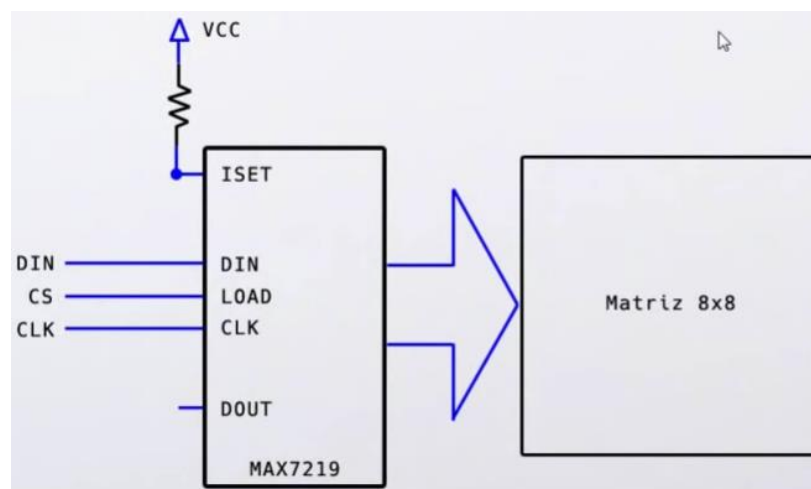


Diagrama de conexión entre un Arduino y una Matriz Led 8X8

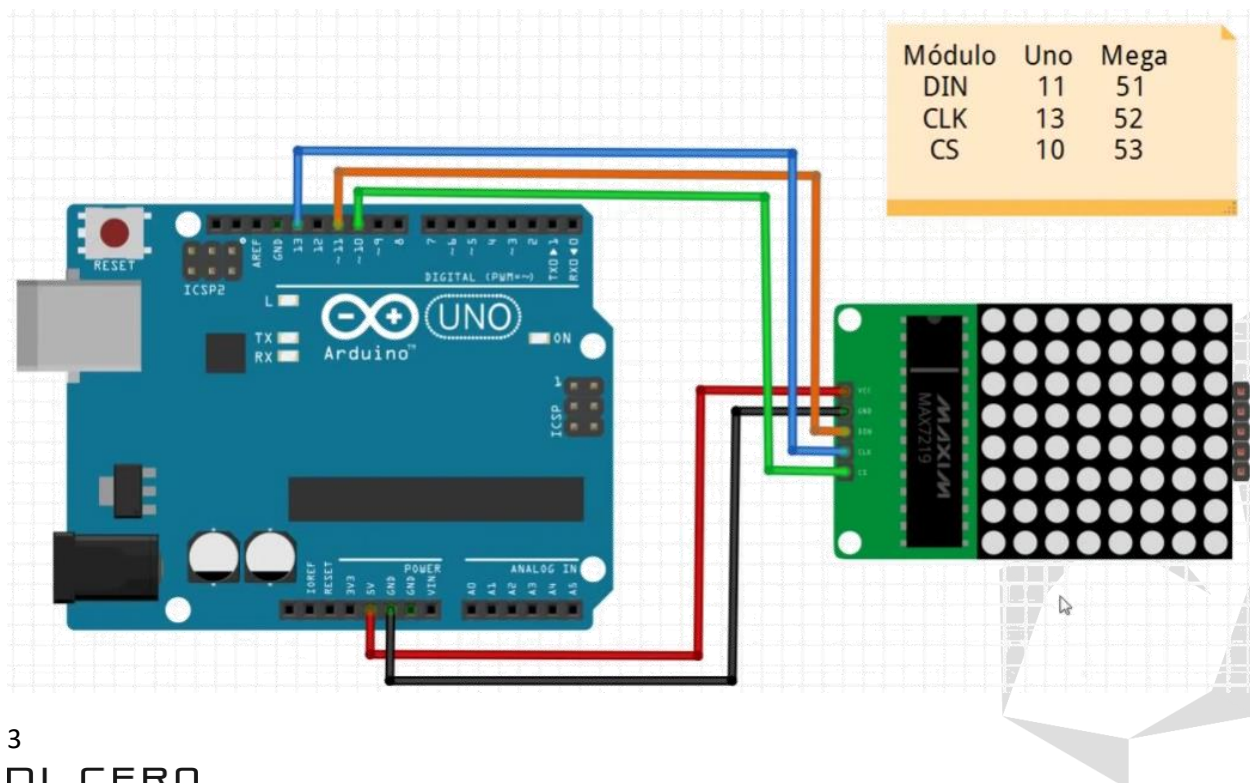
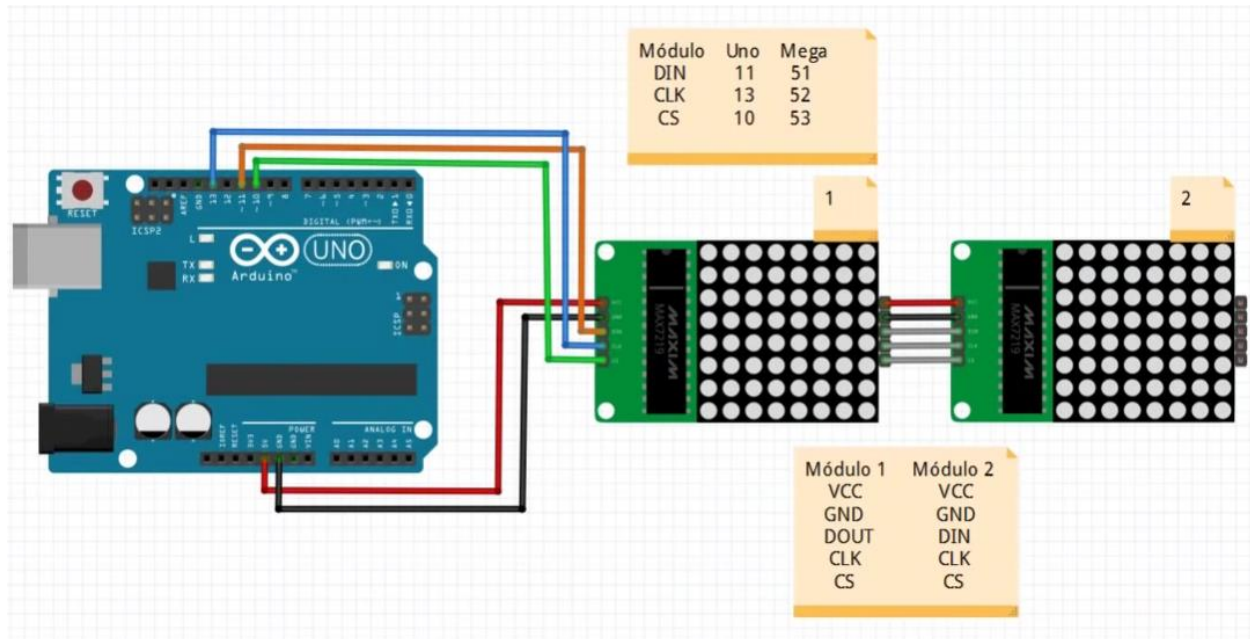


Diagrama de conexión entre un Arduino y dos o más Matrices Led 8X8



Código Arduino Secuencia Encendido y Apagado un Solo Led en Cascada:

```

/*43.1.-Secuencia de encendido y apagado de un solo led que se va recorriendo de fila en fila en una matriz
led de 8X8, encendiendo desde el primer led de la primera fila y columna hasta el último led de la última
fila y columna uno por uno.*/

//IMPORTACIÓN DE LIBRERÍAS
/*Para poder importar una librería debemos dar clic en el botón de:
Programa -> Incluir Librería -> Administrar Bibliotecas -> Buscar: LedControl -> Instalar
Luego ya habiendo realizado eso se debe incluir alguna de las siguientes instrucciones:
#include <NombreLibreria.h>
#include "NombreLibreria.h"*/
#include <LedControl.h> //Librería de control de Matrices led 8X8.

/*Objeto de la clase LedControl, este pide en sus 4 parámetros de su constructor lo siguiente:
- Primer parámetro: Indica el Pin del Arduino al que está conectado el Pin DIN (Master Out
Slave In) de la matriz led 8X8, que indica la dirección de circulación de datos.
- Segundo parámetro: Indica el Pin del Arduino al que está conectado el Pin CLK, que es el
pulso de reloj que marca el paso con el que se mandan los datos del maestro al esclavo.
- Tercer parámetro: Indica el Pin del Arduino al que está conectado el Pin CS (Chip Select),
que habilita la función del circuito integrado MAX7219 que controla a la matriz led 8X8.
- Cuarto parámetro: Indica el número de matrices a controlar, que deben estar conectadas
entre ellas a través del pin DOUT, la librería permite manejar hasta 8 matrices conectadas.*/
LedControl matrizLed = LedControl(11, 13, 10, 1); //DIN = Pin11, CLK = Pin13, CS = Pin10, #MatricesLed.

//CONFIGURACIÓN DE LOS PINES Y LA COMUNICACIÓN SERIAL
void setup() {
  /*LedControl.shutdown(): Método que permite encender la matriz led 8X8, se cuenta el número de
matrices empezando desde cero:
- Primer parámetro: Indica la matriz que se quiere encender, indicando su numeración desde cero.
- Segundo parámetro: Indica que la matriz está encendida con false y que está apagada con true.*/
  matrizLed.shutdown(0, false);
  /*LedControl.setIntensity(): Método que permite indicar el nivel de luminosidad de cada matriz led 8X8,
se cuenta el número de matrices empezando desde cero:
- Primer parámetro: Indica la matriz a la que se le quiere cambiar su nivel de luminosidad, indicando
su numeración desde cero.
- Segundo parámetro: Indica la luminosidad de los leds de la matriz con números de 0 a 15, usualmente
se usa la luminosidad 4 en espacios no tan iluminados o al aire libre.*/
  matrizLed.setIntensity(0, 4);
  /*LedControl.clearDisplay(): Método que limpia cualquier secuencia previamente cargada en la matriz 8X8:
- Primer parámetro: Indica la matriz a la que se le quiere borrar una secuencia pasada, indicando su
numeración desde el número cero.*/
  matrizLed.clearDisplay(0);
}

//EJECUCIÓN DEL PROGRAMA EN UN BUCLE INFINITO: Secuencia de encendido y apagado de un solo led que se va
//recorriendo de fila en fila.
void loop() {
  for(int fila = 0; fila < 8; fila++){
    for(int columna = 0; columna < 8; columna++){
      /*LedControl.setLed(): Método que prende o apaga un led que se encuentre en un número de fila y
columna en específico, para ello se le debe indicar lo siguiente:
- Primer parámetro: Indica la matriz a la que se le quiere encender o apagar un led, exponiendo su

```

```

        numeración desde el número cero.
    - Segundo parámetro: Indica la fila a la que pertenece el led que se quiere encender o apagar.
    - Tercer parámetro: Indica la columna a la que pertenece el led que se quiere encender o apagar.
    - Cuarto parámetro: Indica si se quiere encender con true o apagar con false un led.*/
    matrizLed.setLed(0, fila, columna, true);
    /*delay(ms): Método que detiene la ejecución del programa un cierto tiempo dado en milisegundos.*/
    delay(50); //Detiene el código 50 milisegundos antes de apagar este led y pasar al siguiente.
    matrizLed.setLed(0, fila, columna, false);
    delay(50); //Detiene el código 50 milisegundos antes de encender este led y pasar al siguiente.
}
}
}

```

Código Arduino Secuencia Encendido de Todos Leds de la Matriz:

/*43.2.-Secuencia de encendido de todos los leds de la matriz 8X8, los va recorriendo uno por uno de fila en fila y columna en columna, pero sin apagar el actual antes de pasar al siguiente, esto se repite hasta el último led de la última fila y columna.*/

```

//IMPORTACIÓN DE LIBRERÍAS
/*Para poder importar una librería debemos dar clic en el botón de:
Programa -> Incluir Librería -> Administrar Bibliotecas -> Buscar: LedControl -> Instalar
Luego ya habiendo realizado eso se debe incluir alguna de las siguientes instrucciones:
#include <NombreLibrería.h>
#include "NombreLibrería.h"*/
#include <LedControl.h> //Librería de control de Matrices led 8X8.

/*Objeto de la clase LedControl, este pide en sus 4 parámetros de su constructor lo siguiente:
- Primer parámetro: Indica el Pin del Arduino al que está conectado el Pin DIN (Master Out
Slave In) de la matriz led 8X8, que indica la dirección de circulación de datos.
- Segundo parámetro: Indica el Pin del Arduino al que está conectado el Pin CLK, que es el
pulso de reloj que marca el paso con el que se mandan los datos del maestro al esclavo.
- Tercer parámetro: Indica el Pin del Arduino al que está conectado el Pin CS (Chip Select),
que habilita la función del circuito integrado MAX7219 que controla a la matriz led 8X8.
- Cuarto parámetro: Indica el número de matrices a controlar, que deben estar conectadas
entre ellas a través del pin DOUT, la librería permite manejar hasta 8 matrices conectadas.*/
LedControl matrizLed = LedControl(11, 13, 10, 1); //DIN = Pin11, CLK = Pin13, CS = Pin10, #MatricesLed.

//CONFIGURACIÓN DE LOS PINES Y LA COMUNICACIÓN SERIAL
void setup() {
    /*LedControl.shutdown(): Método que permite encender la matriz led 8X8, se cuenta el número de
    matrices empezando desde cero:
    - Primer parámetro: Indica la matriz que se quiere encender, indicando su numeración desde cero.
    - Segundo parámetro: Indica que la matriz está encendida con false y que está apagada con true.*/
    matrizLed.shutdown(0, false);
    /*LedControl.setIntensity(): Método que permite indicar el nivel de luminosidad de cada matriz led 8X8,
    se cuenta el número de matrices empezando desde cero:
    - Primer parámetro: Indica la matriz a la que se le quiere cambiar su nivel de luminosidad, indicando
    su numeración desde cero.
    - Segundo parámetro: Indica la luminosidad de los leds de la matriz con números de 0 a 15, usualmente
    se usa la luminosidad 4 en espacios no tan iluminados o al aire libre.*/
    matrizLed.setIntensity(0, 8);
    /*LedControl.clearDisplay(): Método que limpia cualquier secuencia previamente cargada en la matriz 8X8:
    - Primer parámetro: Indica la matriz a la que se le quiere borrar una secuencia pasada, indicando su
    numeración desde el número cero.*/
    matrizLed.clearDisplay(0);
}

//EJECUCIÓN DEL PROGRAMA EN UN BUCLE INFINITO: Secuencia de encendido y apagado de un solo led que se va
//recorriendo de fila en fila.
void loop() {
    for(int fila = 0; fila < 8; fila++){
        for(int columna = 0; columna < 8; columna++){
            /*LedControl.setLed(): Método que prende o apaga un led que se encuentre en un número de fila y
            columna en específico, para ello se le debe indicar lo siguiente:
            - Primer parámetro: Indica la matriz a la que se le quiere encender o apagar un led, exponiendo su
            numeración desde el número cero.
            - Segundo parámetro: Indica la fila a la que pertenece el led que se quiere encender o apagar.
            - Tercer parámetro: Indica la columna a la que pertenece el led que se quiere encender o apagar.
            - Cuarto parámetro: Indica si se quiere encender con true o apagar con false un led.*/
            matrizLed.setLed(0, fila, columna, true);
            /*delay(ms): Método que detiene la ejecución del programa un cierto tiempo dado en milisegundos.*/
            delay(50); //Detiene el código 50 milisegundos antes de encender el siguiente.
        }
    }
    /*Al terminar de encender todos los leds de la matriz 8X8, los vuelve a apagar antes de empezar la secuencia
    de nuevo.*/
    matrizLed.clearDisplay(0);
}

```

Código Arduino Animación de Mostrar Números en la Matriz 8X8:

```

/*43.3.-Animación de números mostrada en una matriz led 8X8.*/

//IMPORTACIÓN DE LIBRERÍAS
/*Para poder importar una librería debemos dar clic en el botón de:
Programa -> Incluir Librería -> Administrar Bibliotecas -> Buscar: LedControl -> Instalar
Luego ya habiendo realizado eso se debe incluir alguna de las siguientes instrucciones:
#include <NombreLibrería.h>
#include "NombreLibrería.h"*/

```



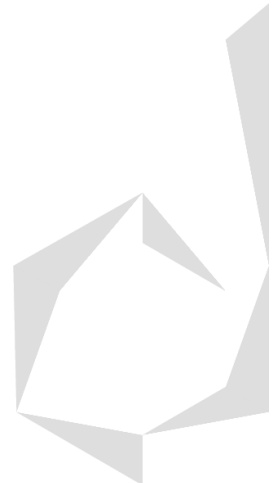
```

#include "LedControl.h" //Libreria de control de Matrices led 8X8.

/*Objeto de la clase LedControl, este pide en sus 4 parámetros de su constructor lo siguiente:
- Primer parámetro: Indica el Pin del Arduino al que está conectado el Pin DIN (Master Out
Slave In) de la matriz led 8X8, que indica la dirección de circulación de datos.
- Segundo parámetro: Indica el Pin del Arduino al que está conectado el Pin CLK, que es el
pulso de reloj que marca el paso con el que se mandan los datos del maestro al esclavo.
- Tercer parámetro: Indica el Pin del Arduino al que está conectado el Pin CS (Chip Select),
que habilita la función del circuito integrado MAX7219 que controla a la matriz led 8X8.
- Cuarto parámetro: Indica el número de matrices a controlar, que deben estar conectadas
entre ellas a través del pin DOUT, la libreria permite manejar hasta 8 matrices conectadas.*/
LedControl matrizLed = LedControl(11, 13, 10, 1); //DIN = Pin11, CLK = Pin13, CS = Pin10, #MatricesLed.

//DECLARACIÓN DE VARIABLES Y CONSTANTES: Los leds de la matriz se encienden con 1 y se apagan con 0.
/*Variable que detiene el programa antes de enseñar un número distinto en la matriz led 8X8.*/
#define demora 500 //Delay de 1000 ms.
/*Variable que representa el número cero.*/
byte cero [8] = {
    B00111000,
    B01000100,
    B01000100,
    B01000100,
    B01000100,
    B01000100,
    B01000100,
    B00111000
};
/*Variable que representa el número uno.*/
byte uno [8] = {
    B00010000,
    B00110000,
    B00010000,
    B00010000,
    B00010000,
    B00010000,
    B00010000,
    B00110000
};
/*Variable que representa el número dos.*/
byte dos[8]= {
    B00111000,
    B01000100,
    B00000100,
    B00000100,
    B00001000,
    B00010000,
    B00100000,
    B01111100
};
/*Variable que representa el número tres.*/
byte tres[8]= {
    B00111000,
    B01000100,
    B00000100,
    B00011000,
    B00000100,
    B00000100,
    B01000100,
    B00111000
};
/*Variable que representa el número cuatro.*/
byte cuatro[8]= {
    B00001000,
    B00011000,
    B00101000,
    B01001000,
    B01001000,
    B01111100,
    B00001000,
    B00001000
};
/*Variable que representa el número cinco.*/
byte cinco[8]= {
    B01111100,
    B01000000,
    B01000000,
    B01111000,
    B00000100,
    B00000100,
    B01000100,
    B00111000
};
/*Variable que representa el número seis.*/
byte seis[8]= {
    B00111000,
    B01000100,
    B01000000,
    B01111000,
    B01000100,
    B01000100,
    B01000100,
    B00111000
};
/*Variable que representa el número siete.*/
byte siete[8]= {

```



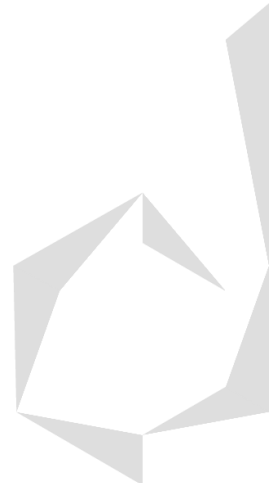

```

B01111100,
B00000100,
B00000100,
B00001000,
B00010000,
B00100000,
B00100000,
B00100000
};
/*Variable que representa el número ocho.*/
byte ocho[8]= {
  B00111000,
  B01000100,
  B01000100,
  B00111000,
  B01000100,
  B01000100,
  B01000100,
  B00111000
};
/*Variable que representa el número nueve.*/
byte nueve[8]= {
  B00111000,
  B01000100,
  B01000100,
  B01000100,
  B00111000,
  B00000100,
  B01000100,
  B00111000
};

//CONFIGURACIÓN DE LOS PINES Y LA COMUNICACIÓN SERIAL
void setup() {
  /*LedControl.shutdown(): Método que permite encender la matriz led 8X8, se cuenta el número de
  matrices empezando desde cero:
  - Primer parámetro: Indica la matriz que se quiere encender, indicando su numeración desde cero.
  - Segundo parámetro: Indica que la matriz está encendida con false y que está apagada con true.*/
  matrizLed.shutdown(0, false);
  /*LedControl.setIntensity(): Método que permite indicar el nivel de luminosidad de cada matriz led 8X8,
  se cuenta el número de matrices empezando desde cero:
  - Primer parámetro: Indica la matriz a la que se le quiere cambiar su nivel de luminosidad, indicando
  su numeración desde cero.
  - Segundo parámetro: Indica la luminosidad de los leds de la matriz con números de 0 a 15, usualmente
  se usa la luminosidad 4 en espacios no tan iluminados o al aire libre.*/
  matrizLed.setIntensity(0, 8);
  /*LedControl.clearDisplay(): Método que limpia cualquier secuencia previamente cargada en la matriz 8X8:
  - Primer parámetro: Indica la matriz a la que se le quiere borrar una secuencia pasada, indicando su
  numeración desde el número cero.*/
  matrizLed.clearDisplay(0);
}

//FUNCIONES PROPIAS UTILIZADAS PARA MOSTRAR CADA NÚMERO:
void mostrar_0() {
  for(int fila = 0; fila < 8; fila++){
    /*LedControl.setRow(): Método que prende o apaga varios leds que muestren un símbolo, número o lo que
    sea, pero para ello antes se debe haber creado una matriz de números binarios donde se cree la figura
    encendiendo leds mandándole a la matriz un 1 lógico y apagándolos con un 0 lógico:
    - Primer parámetro: Indica la matriz a la que se le quiere encender o apagar un led, exponiendo su
    numeración desde el número cero.
    - Segundo parámetro: Indica la fila a la que pertenece el led que se quiere encender o apagar.
    - Tercer parámetro: Indica el array que describe la figura a mostrar, donde se enciende un led con 1
    y se apaga con 0.*/
    matrizLed.setRow(0, fila, cero[fila]);
  }
}
void mostrar_1() {
  for(int fila = 0; fila < 8; fila++){
    matrizLed.setRow(0, fila, uno[fila]);
  }
}
void mostrar_2() {
  for(int fila = 0; fila < 8; fila++){
    matrizLed.setRow(0, fila, dos[fila]);
  }
}
void mostrar_3() {
  for(int fila = 0; fila < 8; fila++){
    matrizLed.setRow(0, fila, tres[fila]);
  }
}
void mostrar_4() {
  for(int fila = 0; fila < 8; fila++){
    matrizLed.setRow(0, fila, cuatro[fila]);
  }
}
void mostrar_5() {
  for(int fila = 0; fila < 8; fila++){
    matrizLed.setRow(0, fila, cinco[fila]);
  }
}
void mostrar_6() {
  for(int fila = 0; fila < 8; fila++){
    matrizLed.setRow(0, fila, seis[fila]);
  }
}

```




```

}
void mostrar_7() {
  for(int fila = 0; fila < 8; fila++){
    matrizLed.setRow(0, fila, siete[fila]);
  }
}
void mostrar_8() {
  for(int fila = 0; fila < 8; fila++){
    matrizLed.setRow(0, fila, ocho[fila]);
  }
}
void mostrar_9() {
  for(int fila = 0; fila < 8; fila++){
    matrizLed.setRow(0, fila, nueve[fila]);
  }
}

//EJECUCIÓN DEL PROGRAMA EN UN BUCLE INFINITO: Secuencia de encendido y apagado de un solo led que se va
//recorriendo de fila en fila.
void loop() {
  for(int fila = 0; fila < 8; fila++){
    for(int columna = 0; columna < 8; columna++){
      /*Uso de la función propia declarada previamente.*/
      mostrar_0();
      /*delay(ms): Método que detiene la ejecución del programa un cierto tiempo dado en milisegundos.*/
      delay(demora); //Detiene el código antes de mostrar el siguiente número.
      mostrar_1();
      delay(demora);
      mostrar_2();
      delay(demora);
      mostrar_3();
      delay(demora);
      mostrar_4();
      delay(demora);
      mostrar_5();
      delay(demora);
      mostrar_6();
      delay(demora);
      mostrar_7();
      delay(demora);
      mostrar_8();
      delay(demora);
      mostrar_9();
      delay(demora);
    }
  }
}

```

Referencias

Bitwise Ar, “Arduino desde cero en Español - Capítulo 43 - Matriz LED 8x8 con MAX7219 y LedControl”, 2018 [Online], Available: https://www.youtube.com/watch?v=_QNUliuW2N0

