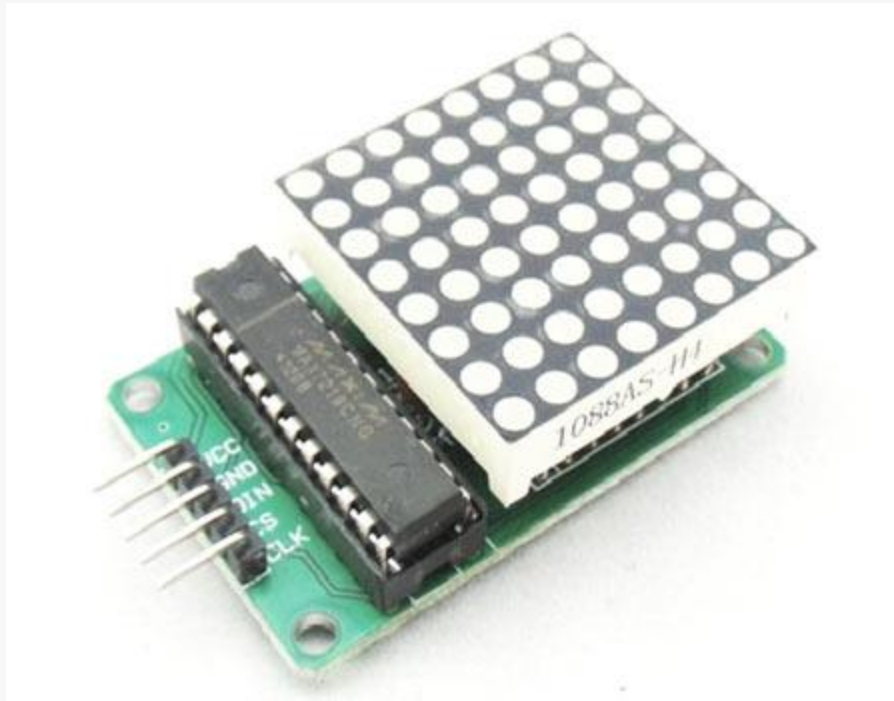


Display Scrolling text on 8x8 LED matrix project



What is 8x8 LED matrix

- An 8 x 8 LED matrix display is used in this project to display the information. LED matrices are available in different styles like single color, dual color, multi-color or RGB LED matrix.
- They are also available in different dimensions like 5 x 7, 8 x 8, 16 x 16, 32 x 32 etc. Based on the arrangement of the LEDs in the matrix, an LED matrix can be either common row anode or common row cathode.
- In case of common row anode type LED matrix, the current sources (high or positive voltage) are given to the rows A-D and the current sinks (low or negative voltage or ground) are given to the columns 1-4.

What is 8x8 LED matrix

- In case of common row cathode type LED matrix, the current sources (high or positive voltage) are given to the columns 1-4 and the current sinks (low or negative voltage or ground) are given to the rows A-D.

- The LED matrix can be driven in two ways. They are parallel (where each row or column are sent with parallel data) and serial (where the data is sent serially and an IC is used to convert this serial data into parallel data).
- MAX 7219 is a common cathode display driver with serial input and parallel output. It is used to interface microprocessors and microcontrollers with 64 individual LEDs (8 x 8 LED matrix for example has 64 LEDs), seven segment LED displays up to 8 digits or bar graph displays.
- The 8 x 8 LED matrix is connected to the MAX 7219 as shown in the circuit diagram and the data input is received from the Arduino board to the MAX 7219.

Working of project

- In this project, there will be a simple interface between Arduino and 8X8 LED Matrix to display information (even scrolling information and images can be displayed).
- Even though a single 8 x 8 LED matrix with corresponding MAX 7219 IC is used in this project, multiple LED matrices can be connected in series for long scrolling display.

The working of the system is as follows:-

- 3 of the 14 available digital input / output pins are used to control the display driver IC MAX 7219. The 3 pins on the MAX7219 IC are clock, data in and load (or cs in case of MAX 7221 IC). A maximum clock frequency of 10MHz can be applied. DIN (Data in) accepts the serial data from the microcontroller or Arduino board.
- It is 16 bit long where the first 8 bits (D0 – D7) are for driving the columns (SEG A-G and DP of the MAX 7219 IC) of the LED matrix and the next 8 bits (D8 – D15) are for driving the (DIG 0-7 of the MAX 7219 IC) rows of the LED matrix.

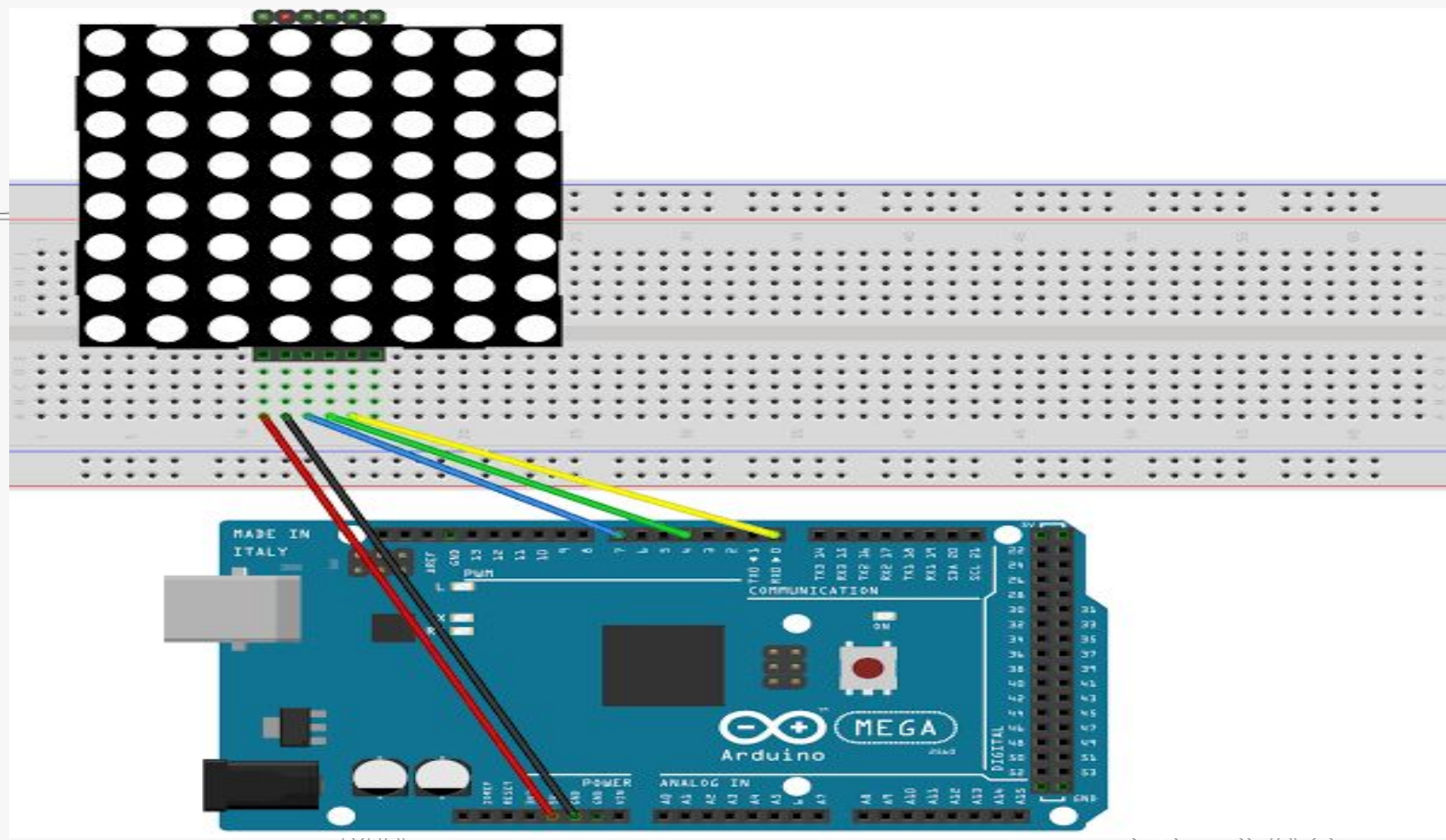
- The load pin (or CS or chip select pin in case of Max 7221 IC) latches the serial input data on its rising edge.
- Another important pin on MAX 7219 is the ISET, which sets the peak current to the segment to drive all the LEDs. It is connected via a resistor (R_1), which is called RSET. The capacitors filters out any noise in the supply.
- When the serial data in is sent using the Arduino (through the program), the serial data is converted into segments and digits to drive columns and rows of the LED matrix. According to the data sent, the corresponding LEDs on the matrix light up and display the message.

- The program written here is for scrolling text display. It might be difficult to view long scrolling data on a single 8 x 8 LED matrix. Hence, multiple LED matrices can be chained to form a long matrix.
- The no. of MAX 7219 ICs are equal to the no. of 8 x 8 LED matrices. In order to extend the display to multiple LED matrices, the Data OUT (DOUT) pin of the first MAX 7219 must be connected to the Data IN (DIN) pin of the second MAX 7219 IC. This process must be continued for multiple LED matrices.
- (The positioning of the LED Matrix in the chain is important. The first LED matrix must be placed at the right of the chain.)

Components Required:

- Arduino Mega
- 8x8 LED Matrix
- USB Cable
- Jumper wires
- BreadBoard

Connection Diagram



Connections

1. Connect Vcc to +5V of Arduino.
2. Connect GND with GND pin of Arduino.
3. Connect DIN with 8 pin of Arduino.
4. Connect CLK with 10 pin of Arduino.
5. Connect CS with 9 pin of Arduino.



Display_scrolling_text_on_led_matrix

```
{0,0,0,0,0,0,0,0,1,0,1,0,1,1,0,1,0,1,0,0,1,1,0,1,0,1,0,1,1,0,1,1,1,0,1,1,1,0,0,1,1,1,0,0,0,0,0,0,0},  
{0,0,0,0,0,0,0,0,1,0,0,1,1,1,0,0,1,0,0,1,0,0,1,0,0,1,0,0,1,1,1,0,0,1,1,1,0,0,1,1,1,0,0,0,0,0,0},  
{0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0,0,1,1,1,0,0,0,0,0,0},  
{0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,1,1,1,1,0,0,0,0,0,0},  
{0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,1,1,1,0,0,0,0,0,0}  
};
```

```
byte firework[18][8] = {  
  {0x00,0x00,0x00,0x80,0x00,0x00,0x00,0x00},  
  {0x00,0x00,0x00,0x00,0x40,0x00,0x00,0x00},  
  {0x00,0x00,0x00,0x20,0x00,0x00,0x00,0x00},  
  {0x00,0x00,0x00,0x00,0x10,0x00,0x00,0x00},  
  {0x00,0x00,0x00,0x08,0x00,0x00,0x00,0x00},  
  {0x00,0x00,0x00,0x08,0x08,0x00,0x00,0x00},  
  {0x00,0x00,0x00,0x08,0x0C,0x00,0x00,0x00},  
  {0x00,0x00,0x00,0x0C,0x0C,0x00,0x00,0x00},  
  {0x00,0x00,0x0A,0x0C,0x0C,0x0A,0x00,0x00},  
  {0x00,0x09,0x0A,0x0C,0x0C,0x0A,0x09,0x00},  
  {0x11,0x09,0x1A,0x0C,0x0C,0x1A,0x09,0x11},  
  {0x22,0x12,0x34,0x18,0x18,0x34,0x12,0x22},  
  {0x00,0x24,0x68,0x30,0x30,0x68,0x24,0x00},  
  {0x00,0x48,0xD0,0x00,0x00,0xD0,0x48,0x00},  
  {0x00,0x90,0xA0,0x00,0x00,0xA0,0x90,0x00},  
  {0x00,0x20,0x00,0x00,0x00,0x00,0x20,0x00},  
  {0x40,0x00,0x00,0x00,0x00,0x00,0x40},  
  {0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00}  
};
```

Display_scrolling_text_on_led_matrix | Arduino 1.8.19

File Edit Sketch Tools Help



Display_scrolling_text_on_led_matrix

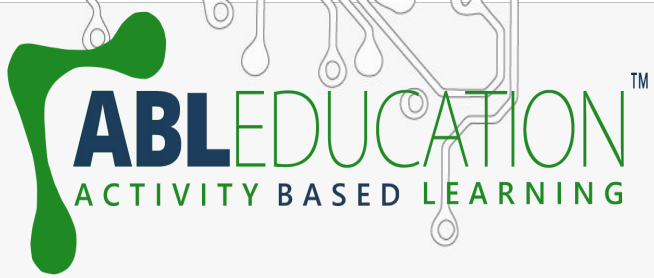
```
    {0x40,0x00,0x00,0x00,0x00,0x00,0x00,0x40},  
    {0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00}  
};
```

```
LedControl lc=LedControl(DIN,CLK,CS,0);
```

```
void setup(){  
  ml.begin(DIN, CLK, CS, 3); // dataPin, clkPin, csPin, number of 8x8 matrix  
  lc.shutdown(0,false); //The MAX72XX is in power-saving mode on startup  
  lc.setIntensity(0,10); // Set the brightness to maximum value  
  lc.clearDisplay(0); // and clear the display  
}
```

```
void loop(){  
  lc.clearDisplay(0);  
  
  PrintAll();  
}
```

```
void PrintAll()  
{  
  //PrintPersianAysan();  
  
  // BeatHeart(50,200);  
  //delay(200);  
  
  //BeatHeart(50,200);  
  //delay(200);  
}
```



Display_scrolling__text_on_led_matrix | Arduino 1.8.19

File Edit Sketch Tools Help



Display_scrolling__text_on_led_matrix

```
//PrintPersianAysan();
```

```
// BeatHeart (50,200);  
//delay(200);
```

```
//BeatHeart (50,200);  
//delay(200);
```

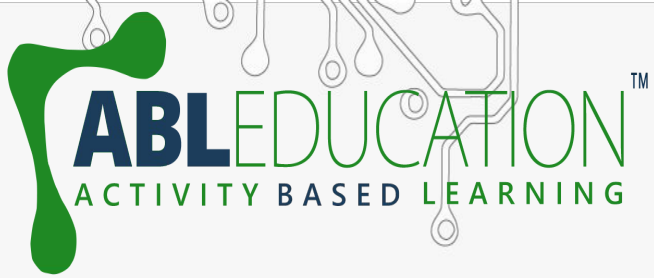
```
//BeatHeart (50,200);  
// delay(500);
```

```
ml.borrar();  
ml.escribirFraseScroll("ABL EDUCATION ", 100); // Text, speed (lower is faster)
```

```
//ml.escribirCaracter('A',0);  
//PrintOneFadeCharacter(30,'A',0);  
//PrintOneFadeCharacter(30,'N',0);  
//PrintOneFadeCharacter(30,'J',0);  
//PrintOneFadeCharacter(30,'A',0);  
//PrintOneFadeCharacter(30,'L',0);  
//PrintOneFadeCharacter(30,'I',0);
```

```
lc.clearDisplay(0);
```

```
//
```

Display_scrolling_text_on_led_matrix | Arduino 1.8.19

File Edit Sketch Tools Help



Display_scrolling_text_on_led_matrix

```
lc.clearDisplay(0);

//Firework(100,200);
delay(1000);

lc.setIntensity(0,10);

//PrintOneFadeCharacter(30,'I',0); PrintOneFadeCharacter(20,Heart3,0);

// lc.clearDisplay(0);
//lc.setIntensity(0,10);

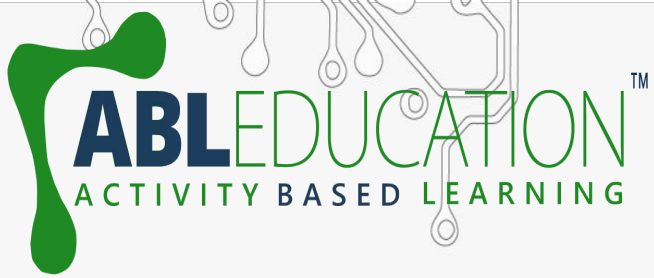
//ml.escribirFraseScroll("You ", 100); // Text, speed (lower is faster)

// lc.clearDisplay(0);

//lc.setIntensity(0,10);

//delay(1000);
}

void BeatHeart(int delay1,int delay2)
{
  lc.setIntensity(0,1);
  printByte(Heart1);
  delay(delay1);
}
```



Display_scrolling__text_on_led_matrix | Arduino 1.8.19

File Edit Sketch Tools Help



Display_scrolling__text_on_led_matrix

```
}  
  
void BeatHeart(int delay1,int delay2)  
{  
  lc.setIntensity(0,1);  
  printByte(Heart1);  
  delay(delay1);  
  lc.setIntensity(0,5);  
  printByte(Heart2);  
  delay(delay1);  
  lc.setIntensity(0,15);  
  printByte(Heart3);  
  delay(delay2);  
  lc.setIntensity(0,5);  
  printByte(Heart2);  
  delay(delay1);  
  lc.setIntensity(0,1);  
  printByte(Heart1);  
  delay(delay1);  
  lc.clearDisplay(0);  
  lc.setIntensity(0,10);  
}  
  
void Firework(int delay1,int delay2)  
{  
  lc.setIntensity(0,1);  
  int exitBrightness = 10;
```



Display_scrolling_text_on_led_matrix

```
}  
  
void Firework(int delay1,int delay2)  
{  
  lc.setIntensity(0,1);  
  int exitBrightness = 10;  
  
  for(int i = 0; i < 18; i++)  
  {  
    printByte(firework[i]);  
  
    if(i == 8)  
      lc.setIntensity(0,3);  
  
    if(i == 9)  
      lc.setIntensity(0,7);  
  
    if(i == 10)  
    {  
      for(int j = 8; j < 16; j++)  
      {  
        lc.setIntensity(0,j);  
        delay(30);  
      }  
    }  
  
    if(i > 11)  
    ,
```



Display_scrolling_text_on_led_matrix | Arduino 1.8.19

File Edit Sketch Tools Help



Display_scrolling_text_on_led_matrix

```
    }

    if(i > 11)
    {
        lc.setIntensity(0,exitBrightness-= 3);
    }

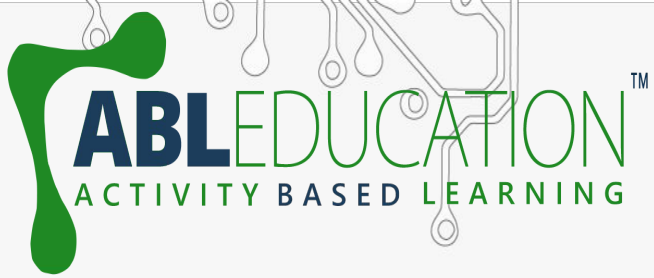
    if(i <= 10)
        delay(delay1);
    else
        delay(delay2);
}

}

void PrintOneFadeCharacter(int fadingDelayMs,byte character[], int initialIntensity)
{
    lc.setIntensity(0,initialIntensity);
    printByte(character);

    for(int i = initialIntensity; i < 15; i++)
    {
        lc.setIntensity(0,i);
        delay(fadingDelayMs);
    }

    for(int i = 15; i >= initialIntensity; i--)
```



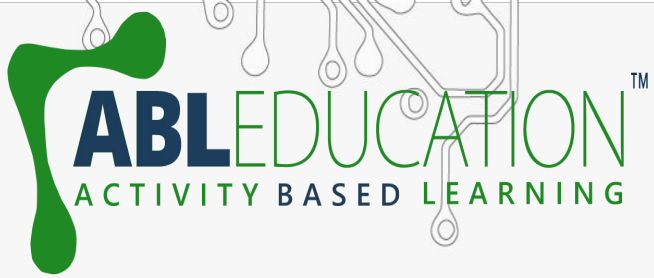
```
Display_scrolling__text_on_led_matrix | Arduino 1.8.19
File Edit Sketch Tools Help
Display_scrolling__text_on_led_matrix
}

for(int i = 15; i >= initialIntensity; i--)
{
  lc.setIntensity(0,i);
  delay(fadingDelayMs);
}

void PrintOneFadeCharacter(int fadingDelayMs,char character, int initialIntensity)
{
  ml.setIntensidad(initialIntensity);
  ml.escribirCaracter(character,0);

  for(int i = initialIntensity; i < 15; i++)
  {
    ml.setIntensidad(i);
    delay(fadingDelayMs);
  }

  for(int i = 15; i >= initialIntensity; i--)
  {
    ml.setIntensidad(i);
    delay(fadingDelayMs);
  }
}
```



Display_scrolling_text_on_led_matrix | Arduino 1.8.19

File Edit Sketch Tools Help



Display_scrolling_text_on_led_matrix

```
void printByte(byte character [])
{
  int i = 0;
  for(i=0;i<8;i++)
  {
    lc.setRow(0,i,character[i]);
  }
}

void PrintPersianTavalodetMobarak ()
{
  ml.clearDisplay(0);

  for(int i = sizeof(TavalodetMobarak[0]); i >= 0; i--)
  {
    boolean frame[8][8];

    for(int ii = 0; ii < 8; ii++)
    {
      for(int j = 0; j < 8 ; j++)
      {
        if(i >= 8)
          frame[ii][j] = TavalodetMobarak[7 - j][i-8+ii];
        else
          frame[ii][j] = 0;
      }
    }
  }
}
```

Display_scrolling_text_on_led_matrix | Arduino 1.8.19

File Edit Sketch Tools Help



Display_scrolling_text_on_led_matrix

```
    if(i >= 8)
        frame[ii][j] = TavalodetMobarak[7 - j][i-8+ii];
    else
        frame[ii][j] = 0;
    }
}

for(int ii = 0; ii < 8; ii++)
{
    for(int jj = 0; jj < 8; jj++)
    {
        ml.setLed(0, ii, jj, frame[ii][jj]);
    }
}
delay(50);
}

void PrintPersianAysan()
{
    ml.clearDisplay(0);
    ml.setIntensity(0,5);

    for(int i = sizeof(PersianAysan2[0]); i >= 0; i--)
    {
        boolean frame[8][8];
```



Display_scrolling_text_on_led_matrix | Arduino 1.8.19

File Edit Sketch Tools Help



Display_scrolling_text_on_led_matrix

```
    }  
}  
  
void PrintPersianAysan()  
{  
  ml.clearDisplay(0);  
  ml.setIntensity(0,5);  
  
  for(int i = sizeof(PersianAysan2[0]); i >= 0; i--)  
  {  
    boolean frame[8][8];  
  
    for(int ii = 0; ii < 8; ii++)  
      for(int j = 0; j < 8; j++)  
        if(i >= 8)  
          frame[ii][j] = PersianAysan2[7 - j][i-8+ii];  
        else  
          frame[ii][j] = 0;  
  
    for(int ii = 0; ii < 8; ii++)  
      for(int jj = 0; jj < 8; jj++)  
        ml.setLed(0, ii, jj, frame[ii][jj]);  
  
    delay(100);  
  }  
}
```


Project Link : <https://youtu.be/jJYlu5onO8w>