2770157 LED Matrix Kit
User Guide

LinkSprite Technologies, Inc.

www.linksprite.com

Revision History

Doc Title	8*8 LED Matrix Shield User Guide	Number	
		Version	2.0

Version	Date	Description	Author
1.0	9/14/2012	First Release	Jack
2.0	11/28/2012		Harry Zhang

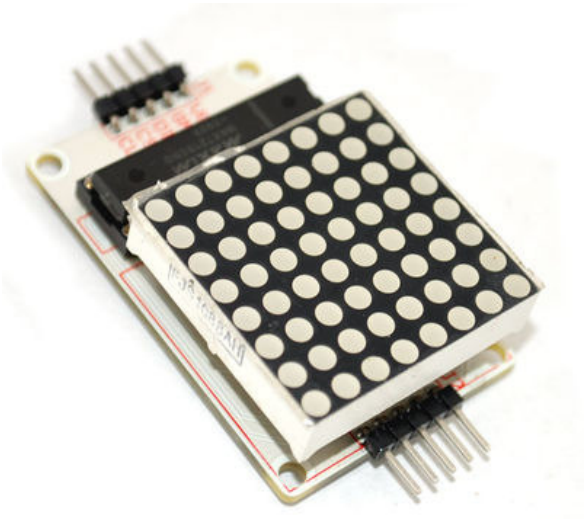
Content

1. Description

1.1 Production Features

This product is a serially driven 8x8 LED Matrix kit powered by MAX7219. It only needs three data lines and two power lines. The 8x8 LED Matrix is easy to use and compatible with Arduino, and its LED brightness adjustment can be implemented in software. This product comes in the form of a kit. It includes the following components:

- MAX7219
- Electrolytic cpaacitor: 10uF/25V
- Resistor: 10K
- Capacitance: 0.1uF
- Headers and receptacles.



8x8 LED Matrix kit has many applications in real life, such as various types of electronic display panels. If the LED matrix is not driven by any peripherals, it’ll waste the interface of devices, and the LED

brightness will be impaired due to insufficient power, so that we cannot get ideal display effect.

The LED matrix can be driven in two ways: parallel or serial. We usually drive it in the serial manner in order to save interface. The serial-driven LED matrix actually dynamically displays the LEDs, i.e., displays the LEDs row-by-row or column-by-column. The persistence of vision for humans is about 0.1s, so as long as we can serially display all 8 rows/columns within 0.1s, we'll see a complete character or pattern.

The 8*8 LED Matrix kit is arranged according to the rows and columns, All the cathode pins of red LED connected together in each column, and each LED's anode pin connected together with the anode pin in same row. In order to drive the LED matrix module, we use a driver chip Max7219 in the module. Set an on-off in a row of the LED matrix module every time by the proper frequency. As mentioned previously, to illumine all the LEDs at the same time visually, actually only 8 LEDs are lighted in each moment. The LED dot matrix has 16 PINs, and we use MAX7219 to drive all the PINs. The 8x8 LED Matrix Shield is controlled by the communication between MAX7219 and Arduino.

1.2 Dimensions

1.3 Specifications

(1) Operating Voltage: DC 4.7V - 5.3V

Typical Voltage: 5V

(2) Operating Current: 320mA

Max Operating Current: 2A

(3) Operating Temperature: 0 ℃ - 50 ℃

Typical Temperature: 25 ℃

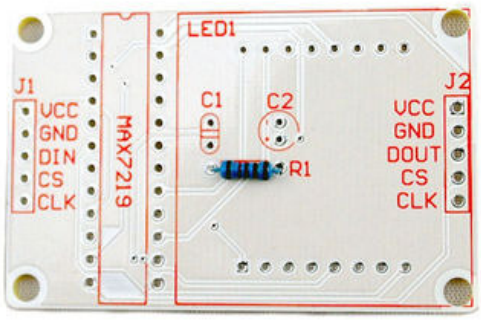
2. Hardware Configurations and Connection Methods

2.1 Hardware Preparation (Use Arduino as microcontroller)

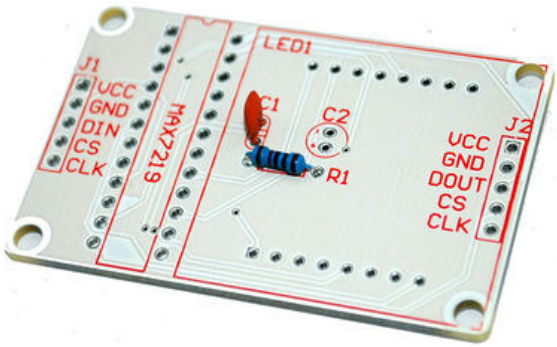
(1) Soldering LED matrix kit

As this product is in a form of a kit, we will need to assemble it. In the following, we are going to show the assembly steps:

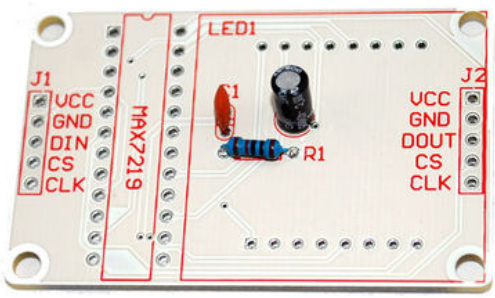
Step 1: Install resistor R1.



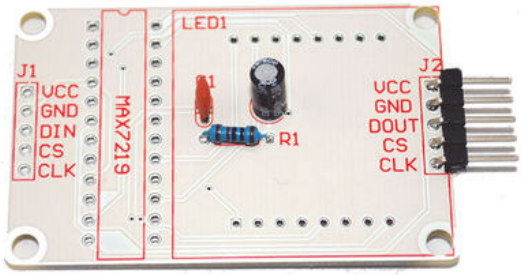
Step 2: Install capacitor C1.



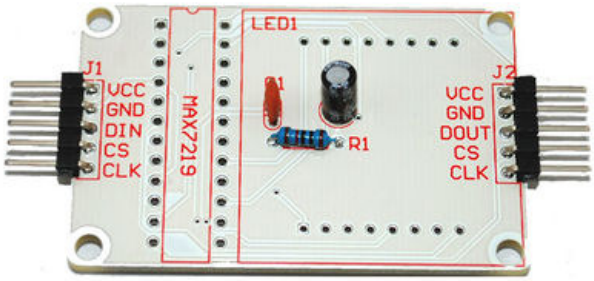
Step 3: Install capacitor C2.



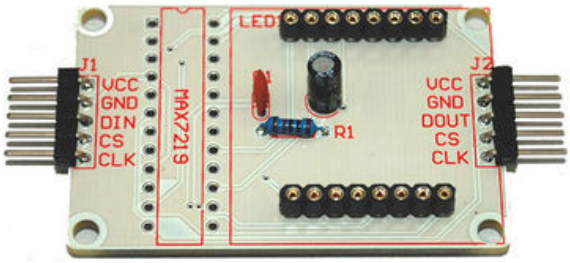
Step 4: Install header J2.



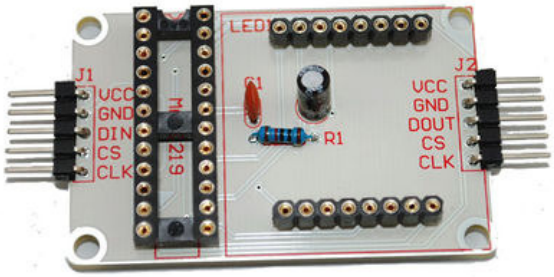
Step 5: Install header J1.



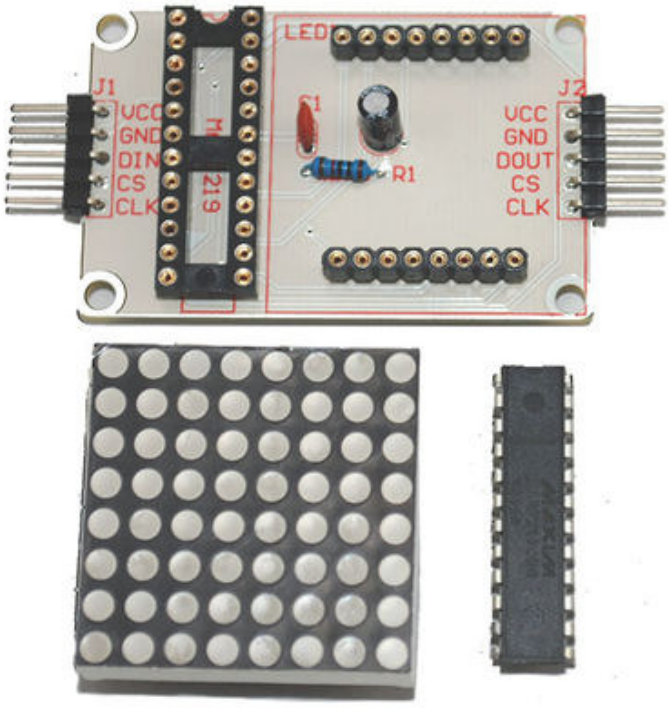
Step 6: Install LED matrix receptacle.



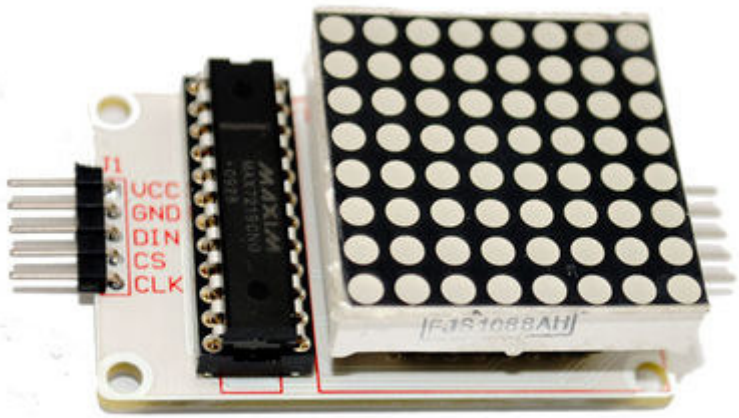
Step 7: Install LED MAX7219 receptacle.



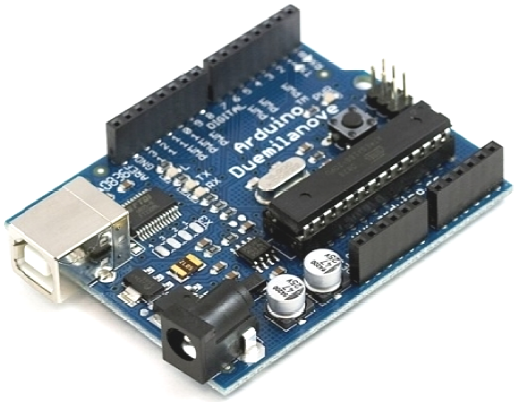
Step 8: Lay down C2, and install the MAX7219 and LED matrix to their receptacles.



Finally, we get the finished LED matrix module:



(2) Arduino



(3) USB Cable

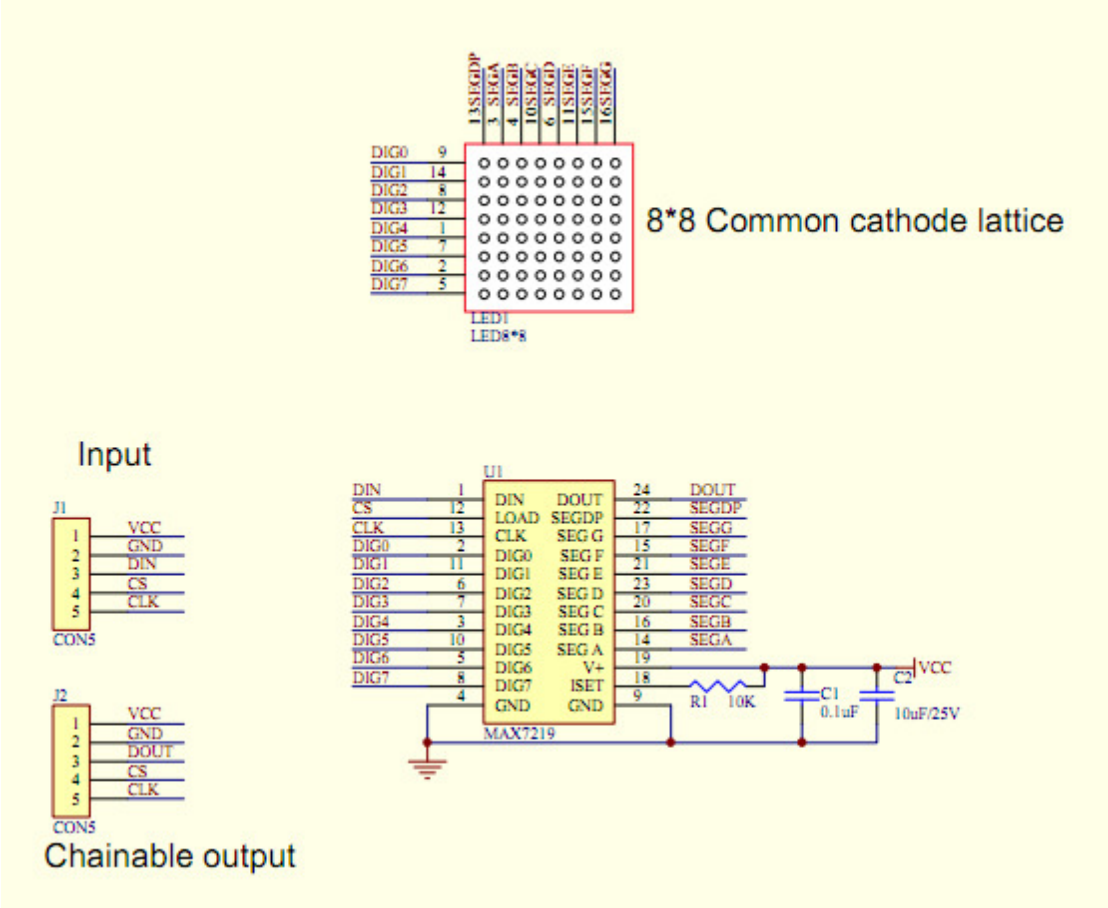
(4) Jumper Wires

(5) Power supply 5V/2A.

2.2 Connection Wires

The schematics of the LED matrix is attached below. Please follow the following instructions to connect hardwares.

(1) The 8x8 LED Matrix must be common-grounded with the Arduino module.
(2) Connect Arduino pin 8 to DIN on the LED Matrix;
(3) Connect Arduino pin 9 to CS on the LED Matrix
(4) Connect Arduino pin 10 to CLK on the LED Matrix
(5) Use independent power supply for the 8x8 LED Matrix Shield, and the supply voltage is 5V/2A.

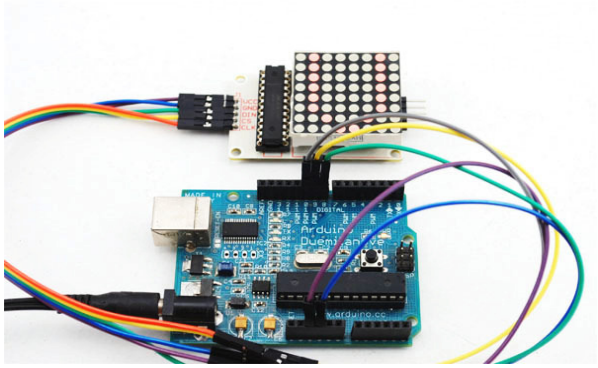


Schematics of the 8x8 LED Matrix

2.3 Testing Steps

(1)Connecting: Connect Arduino pin8 to DIN on 8*8LED Matrix
Connect Arduino pin9 to CS on 8*8LED Matrix
Connect Arduino pin10 to CLK on 8*8LED Matrix

Attention: 8*8LED Matrix must be common-grounded with Arduino module.



(2) Check the corresponding interfaces are properly connected.
(3) Powering up the Arduino demo board with AC adapter.

(4) Observation: The LED matrix should circularly display the digits 0 to 9 on LED screen first, then the characters A to Z, as shown in the above figure.

3. Software

Software preparation

The only necessary software is Arduino 1.0, as shown below.

The timing diagram for Max7219 and its Serial-Data Format are attached below with its Electronic Characteristics.

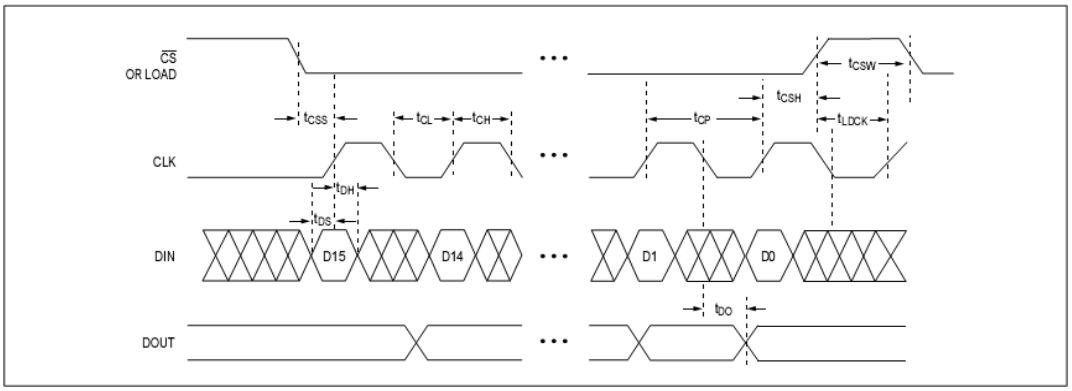


Figure 1. Timing Diagram

Table 1. Serial-Data Format (16 Bits)

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
X	X	X	X	ADDRESS				MSB	DATA						LSB

ELECTRICAL CHARACTERISTICS (continued)

(V+ = 5V ±10%, R_{SET} =9.53kΩ ±1%, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
LOGIC INPUTS						
Input Current DIN, CLK, LOAD, $\overline{CS}$	I_{IH}, I_{IL}	V_{IN} = 0V or V+	-1		1	µA
Logic High Input Voltage	V_{IH}		3.5			V
Logic Low Input Voltage	V_{IL}				0.8	V
Output High Voltage	V_{OH}	DOUT, I_{SOURCE} = -1mA	V+ - 1			V
Output Low Voltage	V_{OL}	DOUT, I_{SINK} = 1.6mA			0.4	V
Hysteresis Voltage	ΔV_I	DIN, CLK, LOAD, $\overline{CS}$		1		V
TIMING CHARACTERISTICS						
CLK Clock Period	t_{CP}		100			ns
CLK Pulse Width High	t_{CH}		50			ns
CLK Pulse Width Low	t_{CL}		50			ns
$\overline{CS}$ Fall to SCLK Rise Setup Time (MAX7221 only)	t_{CSS}		25			ns
CLK Rise to $\overline{CS}$ or LOAD Rise Hold Time	t_{CSH}		0			ns
DIN Setup Time	t_{DS}		25			ns
DIN Hold Time	t_{DH}		0			ns
Output Data Propagation Delay	t_{DO}	C_{LOAD} = 50pF			25	ns
Load-Rising Edge to Next Clock Rising Edge (MAX7219 only)	t_{LDCK}		50			ns
Minimum $\overline{CS}$ or LOAD Pulse High	t_{CSW}		50			ns
Data-to-Segment Delay	t_{DSPD}				2.25	ms

3.2 Example code

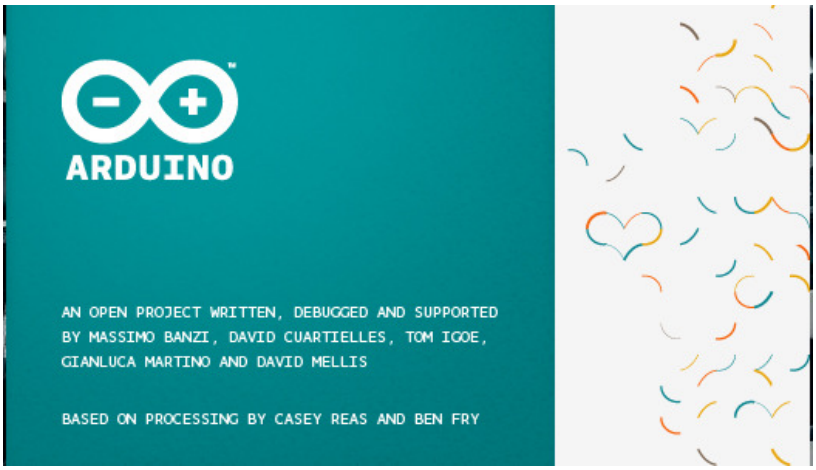
```
 unsigned char i;
 unsigned char j;
/*Port Definitions*/
int Max7219_pinCLK = 10;
int Max7219_pinCS = 9;
int Max7219_pinDIN = 8;

unsigned char disp1[38][8]={
{0x3C,0x42,0x42,0x42,0x42,0x42,0x42,0x3C},//0
{0x10,0x18,0x14,0x10,0x10,0x10,0x10,0x10},//1
{0x7E,0x2,0x2,0x7E,0x40,0x40,0x40,0x7E},//2
{0x3E,0x2,0x2,0x3E,0x2,0x2,0x3E,0x0},//3
{0x8,0x18,0x28,0x48,0xFE,0x8,0x8,0x8},//4
{0x3C,0x20,0x20,0x3C,0x4,0x4,0x3C,0x0},//5
{0x3C,0x20,0x20,0x3C,0x24,0x24,0x3C,0x0},//6
{0x3E,0x22,0x4,0x8,0x8,0x8,0x8,0x8},//7
{0x0,0x3E,0x22,0x22,0x3E,0x22,0x22,0x3E},//8
{0x3E,0x22,0x22,0x3E,0x2,0x2,0x2,0x3E},//9
{0x8,0x14,0x22,0x3E,0x22,0x22,0x22,0x22},//A
{0x3C,0x22,0x22,0x3E,0x22,0x22,0x3C,0x0},//B
{0x3C,0x40,0x40,0x40,0x40,0x40,0x3C,0x0},//C
{0x7C,0x42,0x42,0x42,0x42,0x42,0x7C,0x0},//D
{0x7C,0x40,0x40,0x7C,0x40,0x40,0x40,0x7C},//E
```

```
{0x7C,0x40,0x40,0x7C,0x40,0x40,0x40,0x40},//F
{0x3C,0x40,0x40,0x40,0x40,0x44,0x44,0x3C},//G
{0x44,0x44,0x44,0x7C,0x44,0x44,0x44,0x44},//H
{0x7C,0x10,0x10,0x10,0x10,0x10,0x10,0x7C},//I
{0x3C,0x8,0x8,0x8,0x8,0x8,0x48,0x30},//J
{0x0,0x24,0x28,0x30,0x20,0x30,0x28,0x24},//K
{0x40,0x40,0x40,0x40,0x40,0x40,0x40,0x7C},//L
{0x81,0xC3,0xA5,0x99,0x81,0x81,0x81,0x81},//M
{0x0,0x42,0x62,0x52,0x4A,0x46,0x42,0x0},//N
{0x3C,0x42,0x42,0x42,0x42,0x42,0x42,0x3C},//O
{0x3C,0x22,0x22,0x22,0x3C,0x20,0x20,0x20},//P
{0x1C,0x22,0x22,0x22,0x22,0x26,0x22,0x1D},//Q
{0x3C,0x22,0x22,0x22,0x3C,0x24,0x22,0x21},//R
{0x0,0x1E,0x20,0x20,0x3E,0x2,0x2,0x3C},//S
{0x0,0x3E,0x8,0x8,0x8,0x8,0x8,0x8},//T
{0x42,0x42,0x42,0x42,0x42,0x42,0x22,0x1C},//U
{0x42,0x42,0x42,0x42,0x42,0x42,0x24,0x18},//V
{0x0,0x49,0x49,0x49,0x49,0x2A,0x1C,0x0},//W
{0x0,0x41,0x22,0x14,0x8,0x14,0x22,0x41},//X
{0x41,0x22,0x14,0x8,0x8,0x8,0x8,0x8},//Y
{0x0,0x7F,0x2,0x4,0x8,0x10,0x20,0x7F},//Z
};



void Write_Max7219_byte(unsigned char DATA)
{
          unsigned char i;
        digitalWrite(Max7219_pinCS,LOW);
        for(i=8;i>=1;i--)
         {
           digitalWrite(Max7219_pinCLK,LOW);
           digitalWrite(Max7219_pinDIN,DATA&0x80);// Extracting a bit data
           DATA = DATA<<1;
           digitalWrite(Max7219_pinCLK,HIGH);
         }
}


void Write_Max7219(unsigned char address,unsigned char dat)
{
       digitalWrite(Max7219_pinCS,LOW);
       Write_Max7219_byte(address);          //address, code of LED
       Write_Max7219_byte(dat);             //data, figure on LED
```

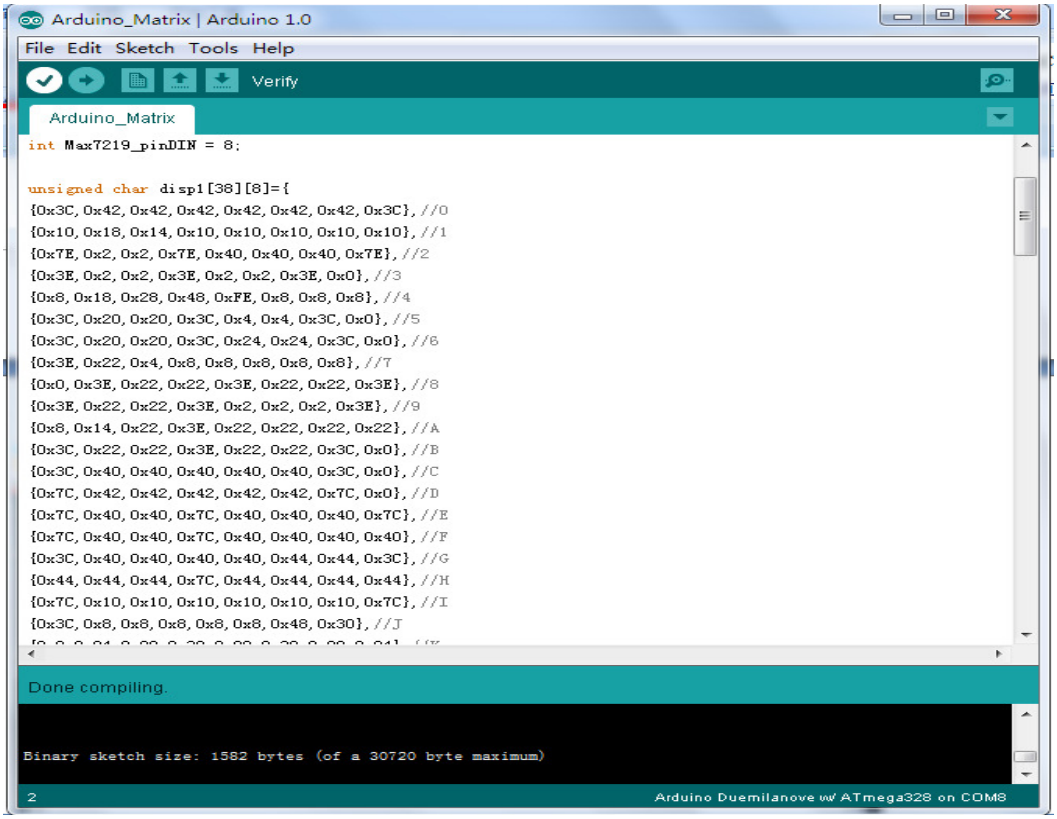
```
       digitalWrite(Max7219_pinCS,HIGH);
}

void Init_MAX7219(void)
{
 Write_Max7219(0x09, 0x00);       //decoding : BCD
 Write_Max7219(0x0a, 0x03);       //brightness
 Write_Max7219(0x0b, 0x07);       //scanlimit; 8 LEDs
 Write_Max7219(0x0c, 0x01);       //power-down mode: 0, normal mode: 1
 Write_Max7219(0x0f, 0x00);       //test display: 1; EOT, display: 0
}



void setup()
{

  pinMode(Max7219_pinCLK,OUTPUT);
  pinMode(Max7219_pinCS,OUTPUT);
  pinMode(Max7219_pinDIN,OUTPUT);
  delay(50);
  Init_MAX7219();
}


void loop()
{
   for(j=0;j<38;j++)
  {
   for(i=1;i<9;i++)
    Write_Max7219(i,disp1[j][i-1]);
   delay(500);
  }
}
```

4. Demo

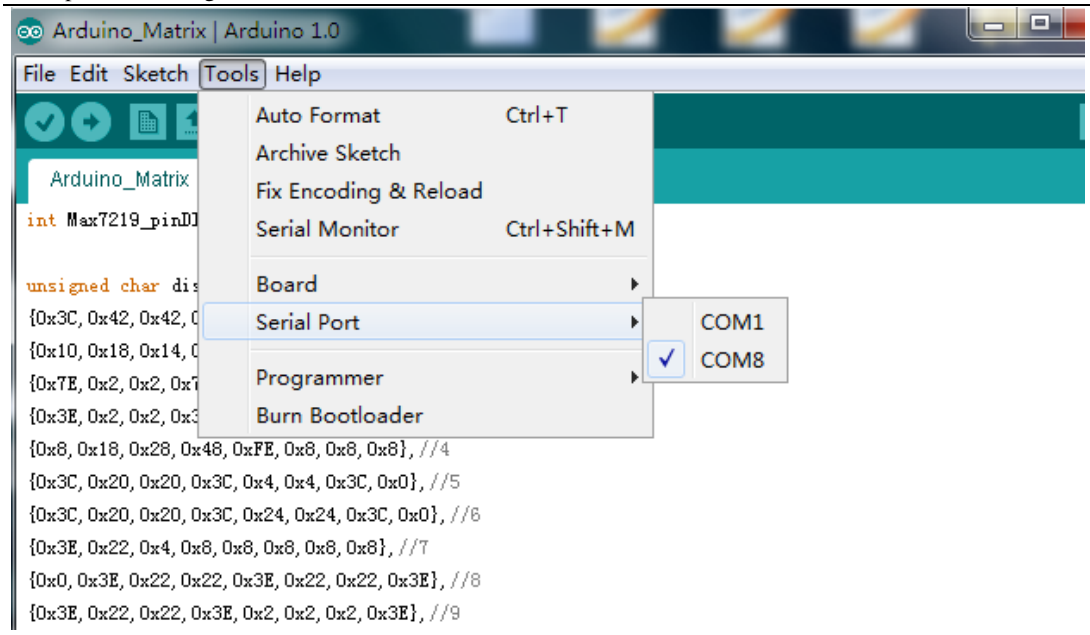
(1) Open Arduino development environment.

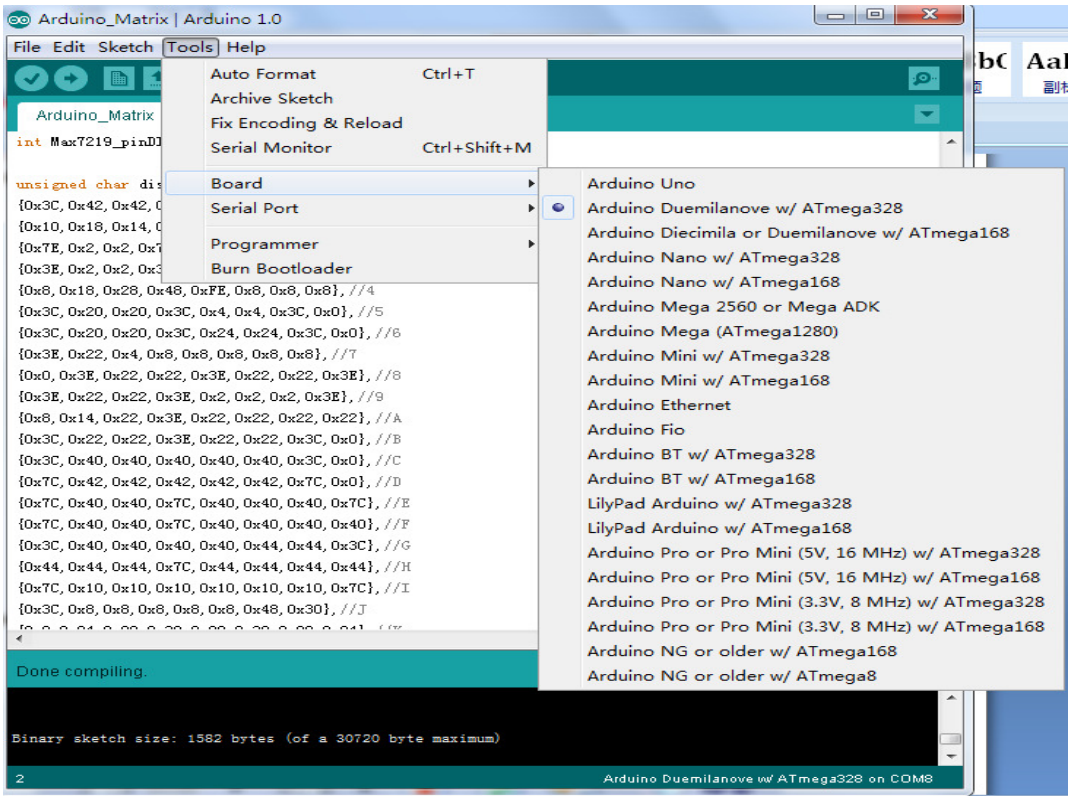


(2) Copy the source code we provide into Arduino compiler, and compile them.

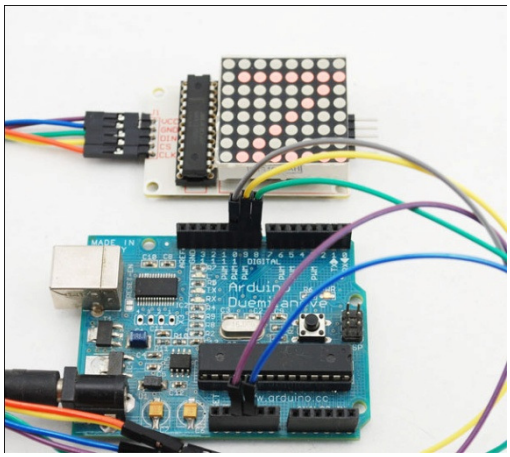


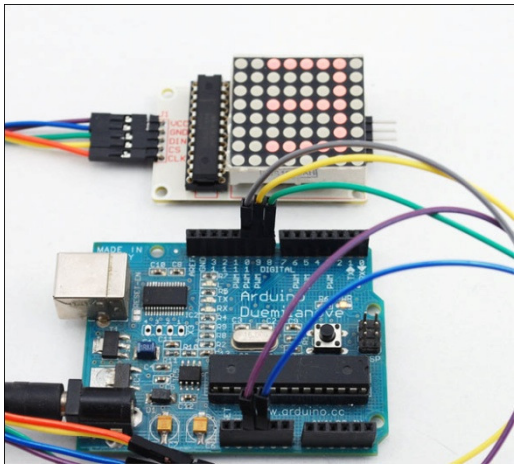
(3) Select proper serial port and board.

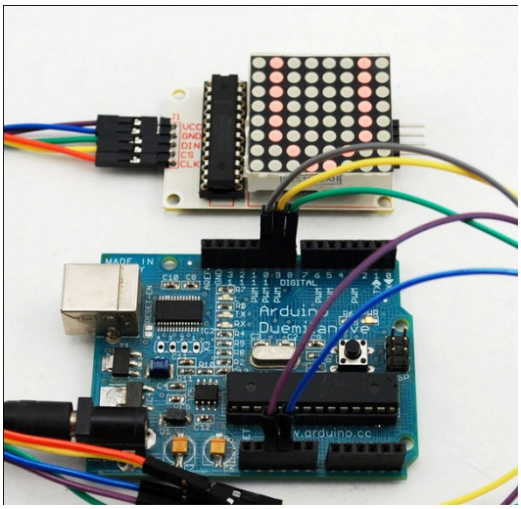




(4) Connect pins according to the schematics, and download codes into Arduino board. You'll see the LED matrix circularly display the digits 0-9 and the characters A-Z.







Add：1067 S Hover St, Unit E-186,Longmont, CO 80501

Tel： 720-204-8599 (MST: 9:00am-5:00pm)

720-279-0774 (MST: 6:00pm- 3:00am)

Support System：www.linksprite.com/support/

Technical Support：support@linksprite.com

sales：sales@linksprite.com

Website：www.linksprite.com