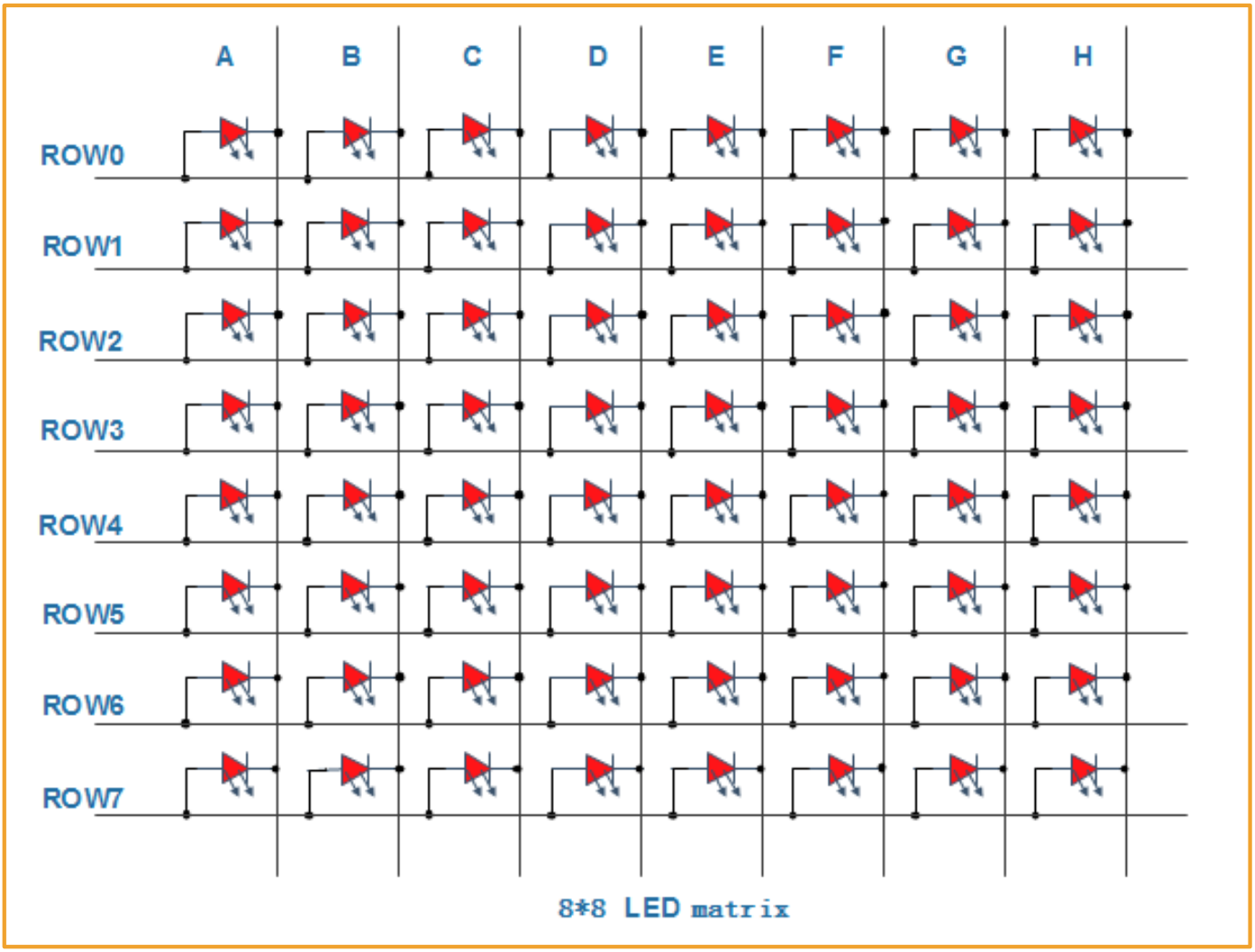
8x8 Dot-matrix Expeiment

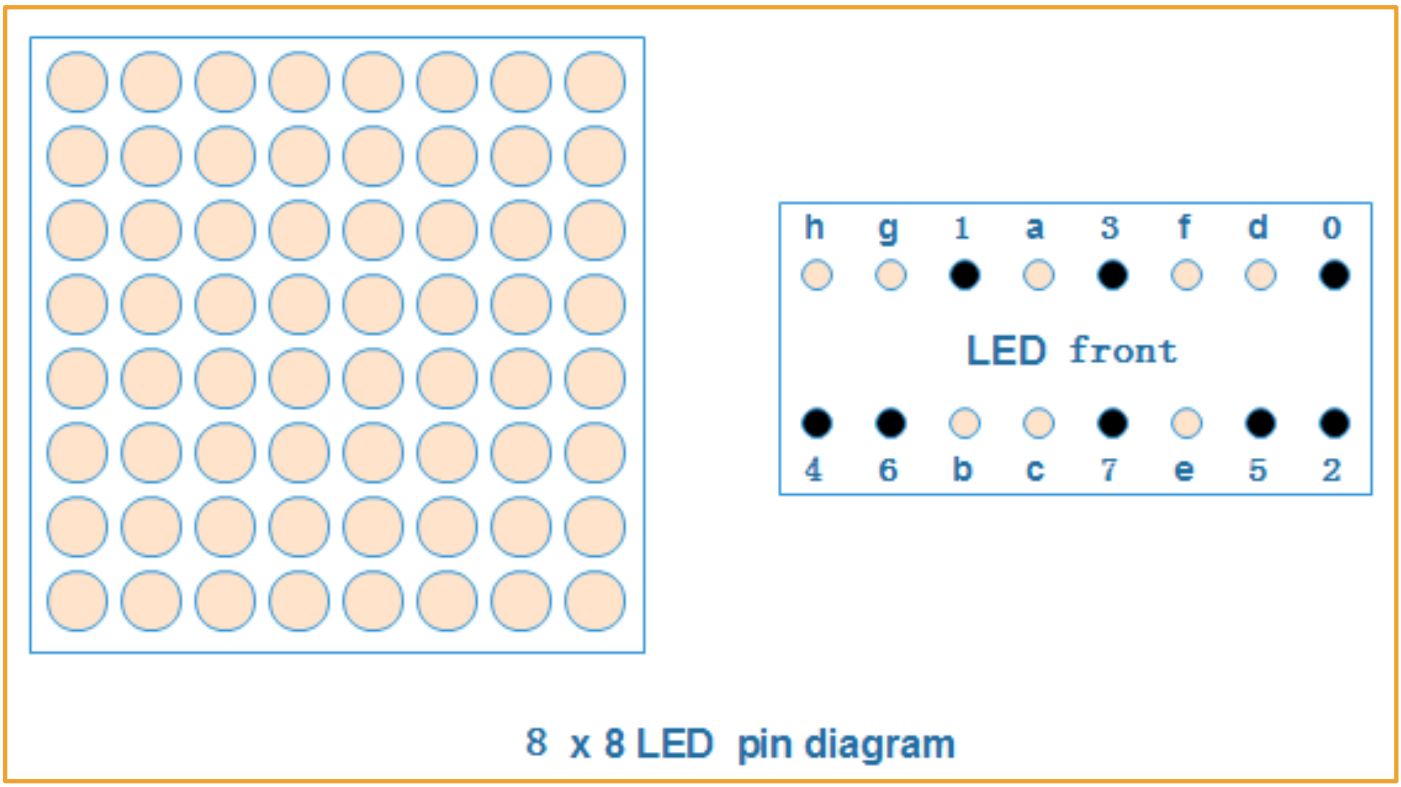
Introduction to 8x8 Dot-matrix

Now it is not difficult for us to display figures after learning the display of Nixie tube. However, if we want to show a variety of patterns in practice, Nixie tube clearly suffers from inadequate capacity, it requires LED dot-matrix. When you walk in the streets, all kinds of LED neon billboards you see are but N x N dot-matrix. Now let's take a look of the internal principle of 8*8 dot-matrix.

Schematic Diagram of 8*8 dot-matrix



Pins of 8*8 dot-matrix



8 x 8 LED pin diagram

The graph displays the appearance of 8 X 8 LED dot-matrix and its pins, and the equivalent circuit is shown in figure (1). As long as its X, Y axes are forward biased, the corresponding LEDs will be lightened. For example, if you want to keep the top left LED lighting, just set ROW0 = 1, A = 0. Due to the through-current of LED is low in practice, if the driving voltage of Arduino is 5v, then we need to connect 1k resistor to the ROW pin.

Scanning of 8*8 dot-matrix

LED commonly displays through scanning , divided into three ways in practice:

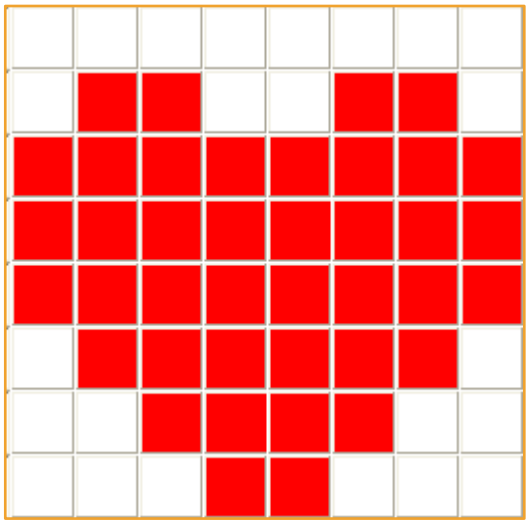
- Spot scanning
- Row scanning
- Column scanning

LED dot-matrix does not require to light up LED one by one, column scanning is more appropriate due to it is attributed to common cathode LED. First, we set the corresponding level of the first column according to the display values. The first column of A will light up when given to low level, the second and the third column will follow if we do the same, then repeating the loop. Now the LED image can be seen due to the visual residual effect of human eyes.

Application of 8*8 led dot-matrix

The internal structure and appearance of dot-matrix are as follows. 8x8 dot-matrix consists of 64 light-emitting diodes, and each light emitting diode lies in the intersection of row and column. When the corresponding row is high level and the column is low level, the corresponding diode is will be bright. If we want to light up the first diode,we need to set the 9th pin at high level and the 11th at low level; If we want to the first row to light, then the 9th pin needs to be at high level and the (13, 3, 4, 10, 6, 11, 15, 16) pins are low level, then the first line will light up; as to the first column, the 13th pin requires to be low level and (9, 14, 8, 12, 1, 7, 2, 5) pins are at high level, then the column will light up.

We try to display a heart-shaped figure in the experiment, so we set the red part at high level and the other at low level, The LED will show a heart-shaped figure after scanning dynamically.



```
char HeartMap[8][8] =
{
    0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 ,
    0 , 1 , 1 , 0 , 0 , 1 , 1 , 0 ,
    1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 ,
    1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 ,
    1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 ,
    0 , 1 , 1 , 1 , 1 , 1 , 1 , 0 ,
    0 , 0 , 1 , 1 , 1 , 1 , 0 , 0 ,
    0 , 0 , 0 , 1 , 1 , 0 , 0 , 0 ,
};
```

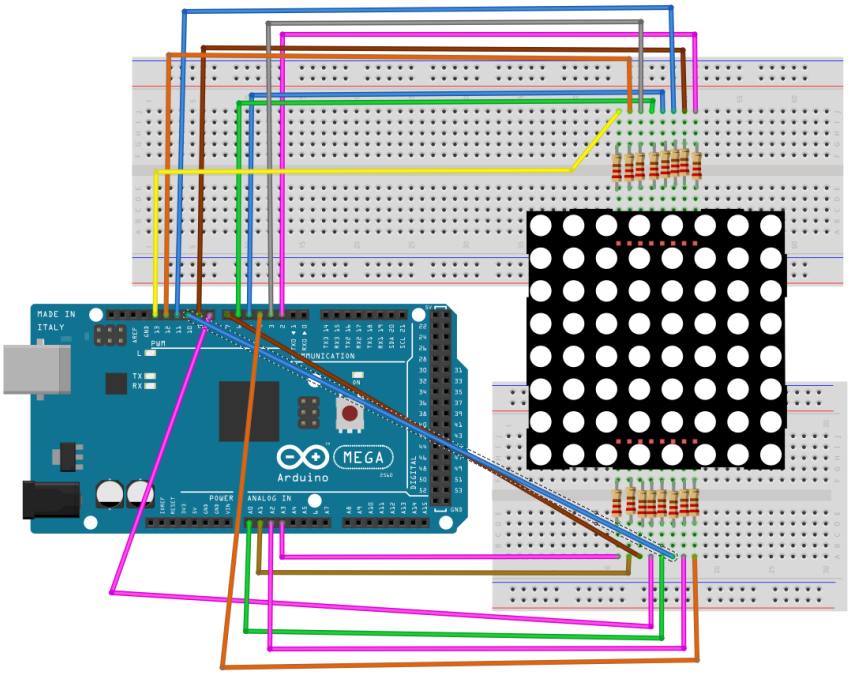
Because it is dynamic scanning. We need to pay attention to two points ghosting and flashing .We need to give each column in the process of scanning the corresponding level, and the common cathode port pull low level, To scan the next column digital tube, we need to pull previous column all low level before. Due to the column scanning, residual effect to human eyes are 25 Hz. But usually the effect is better if sweep frequency is 50Hz, so each column of the delay time can't be more than 1000/50/8 = 2.5ms. If we set the delay to 2ms, effect is better.

Component List

- Keywish Arduino UNO R3 Mainboard
- Breadboard
- USB cable
- 8x8 Dot-matrix * 1
- 1k Resistor * 8
- 4.7k Resistor * 8
- Several jumper wires

Wiring of Circuit

Arduino Mega 2560	8x8 Dot-matrix
2	0
9	d
11	f
5	3
6	a
3	1
12	g
13	h
4	2
A2	5
10	e
A0	7
8	c
7	b
A1	6
A3	4

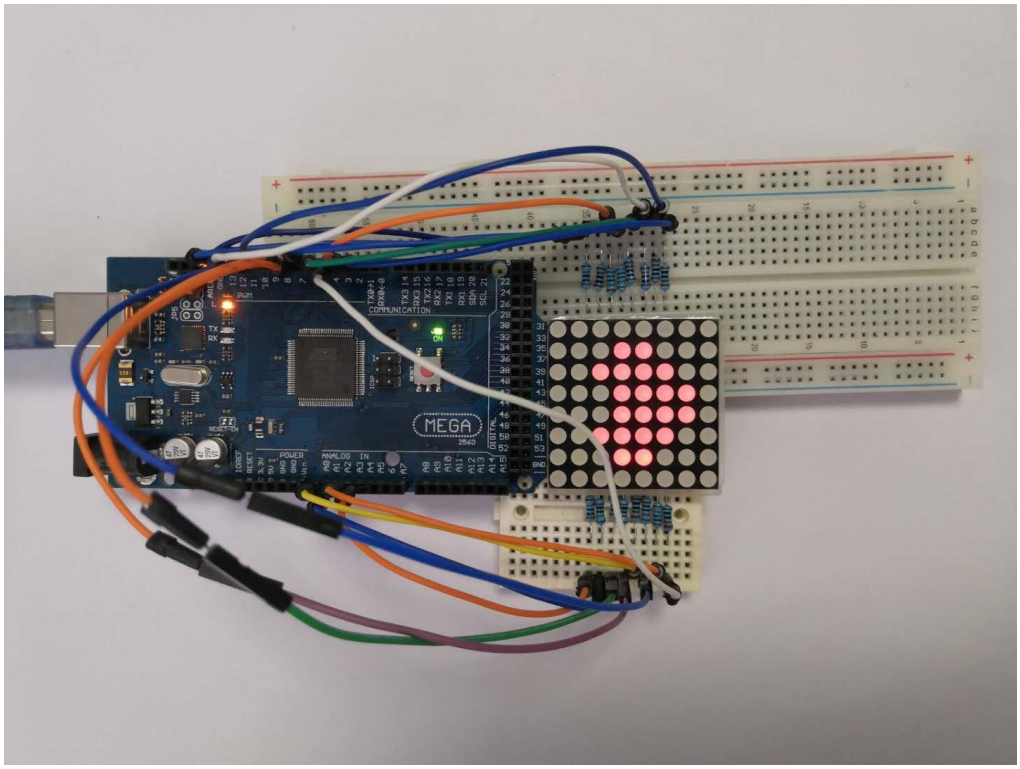


CODE

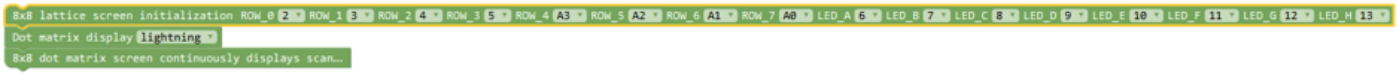
```
#define    ROW_0   2    // definition ROW_ pin 9
#define    ROW_1   3    // definition ROW_ pin 14
#define    ROW_2   4    // definition_ pin 8
#define    ROW_3   5    // definition_ pin 12
#define    ROW_4   A3   // definition_ pin 1
#define    ROW_5   A2   // definition_ pin 7
#define    ROW_6   A1   // definition_ pin 2
#define    ROW_7   A0   // definition_ pin 5
#define     LED_A   6   // definition COL_ pin 13
#define     LED_B   7   // definition COL_ pin 3
#define     LED_C   8   // definition COL_ pin 4
#define     LED_D   9   // definition COL_ pin 10
#define     LED_E   10  // definition COL_ pin 6
#define     LED_F   11  // definition COL_ pin 11
#define     LED_G   12  // definition COL_ pin 15
#define     LED_H    13  // definition COL_ pin 16

const char ROW_PIN[8] =
       {ROW_0, ROW_1, ROW_2, ROW_3, ROW_4, ROW_5, ROW_6, ROW_7};
const char COL_PIN[8] =
       {LED_A, LED_B, LED_C, LED_D, LED_E, LED_F, LED_G, LED_H};
```

```
char HeartMap[8][8] = {
0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 ,
0 , 1 , 1 , 0 , 0 , 1 , 1 , 0 ,
1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 ,
1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 ,
1 , 1 , 1 , 1 , 1 , 1 , 1 , 1 ,
0 , 1 , 1 , 1 , 1 , 1 , 1 , 0 ,
0 , 0 , 1 , 1 , 1 , 1 , 0 , 0 ,
0 , 0 , 0 , 1 , 1 , 0 , 0 , 0 ,};

void setup()
{
    int i = 0 ;
    Serial.begin(115200); // Set the serial port baud rate to 115200
    for(i = 0; i < 8; i++) // Set the port to output mode
    {
        pinMode(ROW_PIN[i] , OUTPUT );
        pinMode(COL_PIN[i] , OUTPUT );
        delay(10);
        digitalWrite(ROW_PIN[i] ,LOW);
        digitalWrite(COL_PIN[i] ,HIGH);
    }
    digitalWrite(ROW_4,LOW); // Initialize ROW_4 to low level
}

void loop()
{
     int i  j;
     for (j = 0; j < 8; j++) // Light up the dot matrix
     {
        for (i = 0; i < 8; i++)
        {
           if (HeartMap[i][j])
               digitalWrite(ROW_PIN[i], HIGH);
           else
               digitalWrite(ROW_PIN[i], LOW);
        }
        digitalWrite(COL_PIN[j], LOW);
        delay(2);
        digitalWrite(COL_PIN[j], HIGH);
    }
}
```

Experiment Result



MIxly programming program



MagicBlock graphical programming program

The 8 ×8 lattice experiment program written by MagicBlock is shown below: