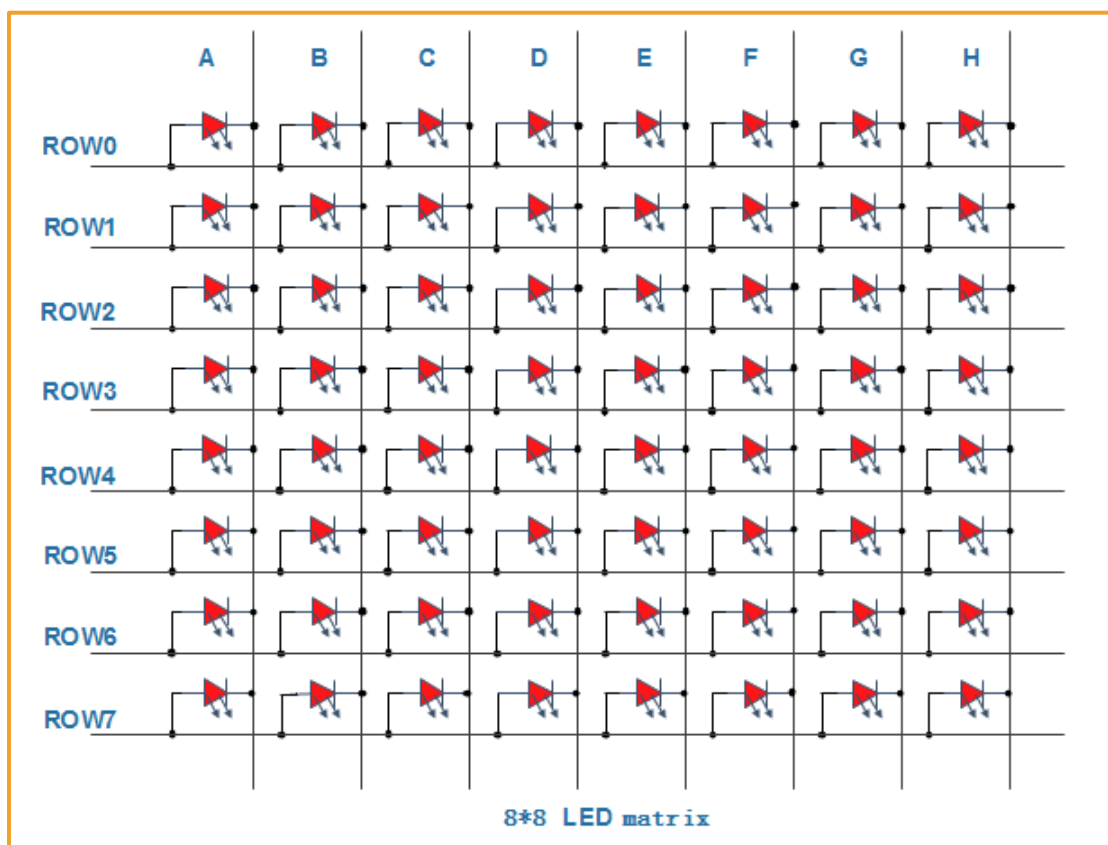


8 × 8 матрицы Эксперимент

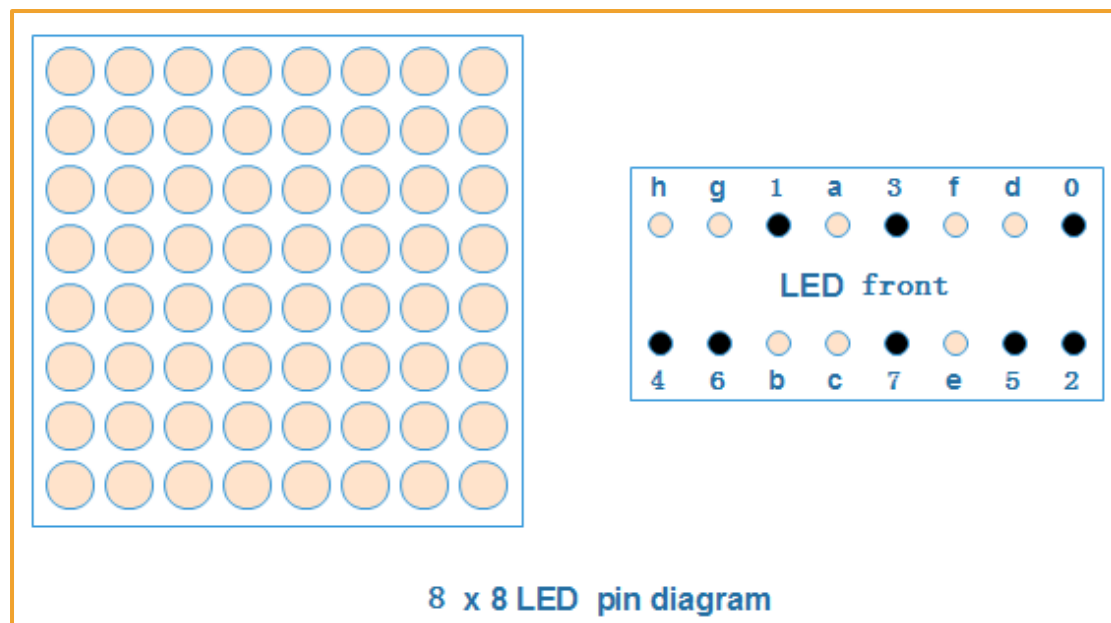
Введение

Теперь, когда мы изучили отображение цифровой трубки, цифровой дисплей не является сложным. Однако, если мы хотим продемонстрировать на практике разнообразные шаблоны, очевидно, что у цифровой трубки есть проблема недостаточной емкости, для чего требуется матрица со светодиодными точками. Когда вы идете по улице, светодиодные неоновые вывески, которые вы видите, представляют собой просто матрицу из $N \times N$ точек. Теперь давайте посмотрим на внутренний принцип матрицы 8×8 .

Матричная схема 8 * 8 точек



Пин 8 * 8 точечная матрица



На рисунке показан внешний вид точечной матрицы светодиодов 8 X 8 и ее выводов. Эквивалентная схема показана на рисунке (1). Пока оси X и Y смещены вперед, загорится соответствующий светодиод. Например, если вы хотите оставить светодиоды в верхнем левом углу, просто установите ROW0 = 1, A = 0. Поскольку фактический ток светодиода низкий, если напряжение привода Arduino составляет 5 В, необходимо подключить резистор 1 кОм к контакту линии.

Сканирование 8 * 8 точечной матрицы

Светодиоды обычно отображаются сканированием и на практике делятся на три метода:

- Точечное сканирования
- Непрерывное сканирование
- сканирование столбцов

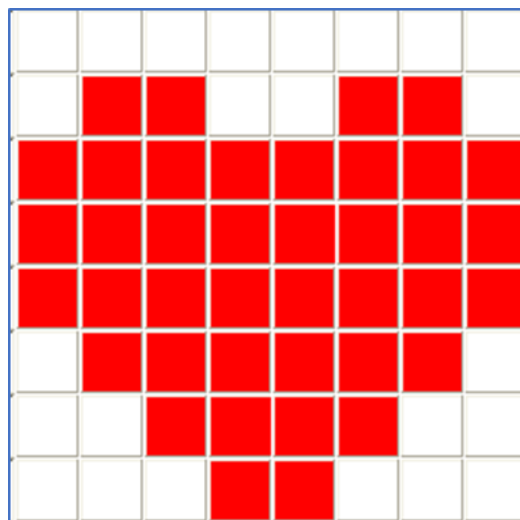
Точечная светодиодная матрица не должна подсвечивать светодиоды один за другим, более подходящим является сканирование столбцов, поскольку оно относится к обычным катодным светодиодам. Сначала мы устанавливаем соответствующий уровень первого столбца в соответствии с отображаемым значением. Первый столбец A загорится, когда он низкий. Если мы сделаем то же самое, второй и третий столбцы загорятся, а затем повторите этот цикл. Теперь светодиодное изображение можно увидеть благодаря остаточному визуальному эффекту человеческого глаза.

Применение 8 * 8 светодиодной матрицы

Внутренняя структура и внешний вид точечной матрицы следующие. Точечная матрица 8x8 состоит из 64 светодиодов, каждый из которых расположен на пересечении строк и столбцов. Когда

соответствующая строчка высокая, а столбец низкий, соответствующий диод светится. Если мы хотим зажечь первый диод, нам нужно установить вывод 9 на высоком уровне и вывод 11 на низком уровне, если мы хотим, чтобы первая линия загорелась, то вывод 9 должен быть высоким уровнем и контакты (13, 3, 4, 10, 6, 11, 15, 16) находятся на низком уровне, тогда будет гореть первый ряд, для первого столбца 13-й вывод должен быть низким, (9,14,8,12,1,7,2,5) вывод высокого уровня, затем столбец загорается.

Мы попытались отобразить график в форме сердца в эксперименте, поэтому мы установили красную часть на высокий уровень, а другую часть на низкий уровень. После того, как светодиод динамически сканирует, будет отображен график в форме сердца.



```
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 ,
};
```

Потому что это динамическое сканирование. Нам нужно обратить внимание на призрак и мерцание двух точек. Нам нужно дать каждому столбцу соответствующий уровень во время процесса сканирования и вывести обычно используемый катодный порт на низкое значение. Чтобы сканировать следующий столбец цифровых трубок, нам нужно вывести предыдущий столбец на низкий уровень. Из-за сканирования колонки остаточное воздействие на человеческий глаз

составляет 25 Гц. Но частота развертки обычно составляет 50 Гц, поэтому время задержки каждого столбца не может превышать $1000/50/8 = 2,5$ мс. Если мы установим задержку на 2 мс, эффект будет лучше

● Компоненты

- Материнская плата Keywish Arduino UNO R3
- Схема макета
- USB-кабель
- 8x8Матрица точек* 1
- 1крезистор* 8
- 4.7ксопротивление* 8
- Несколько перемычек
- Принципиальная схема

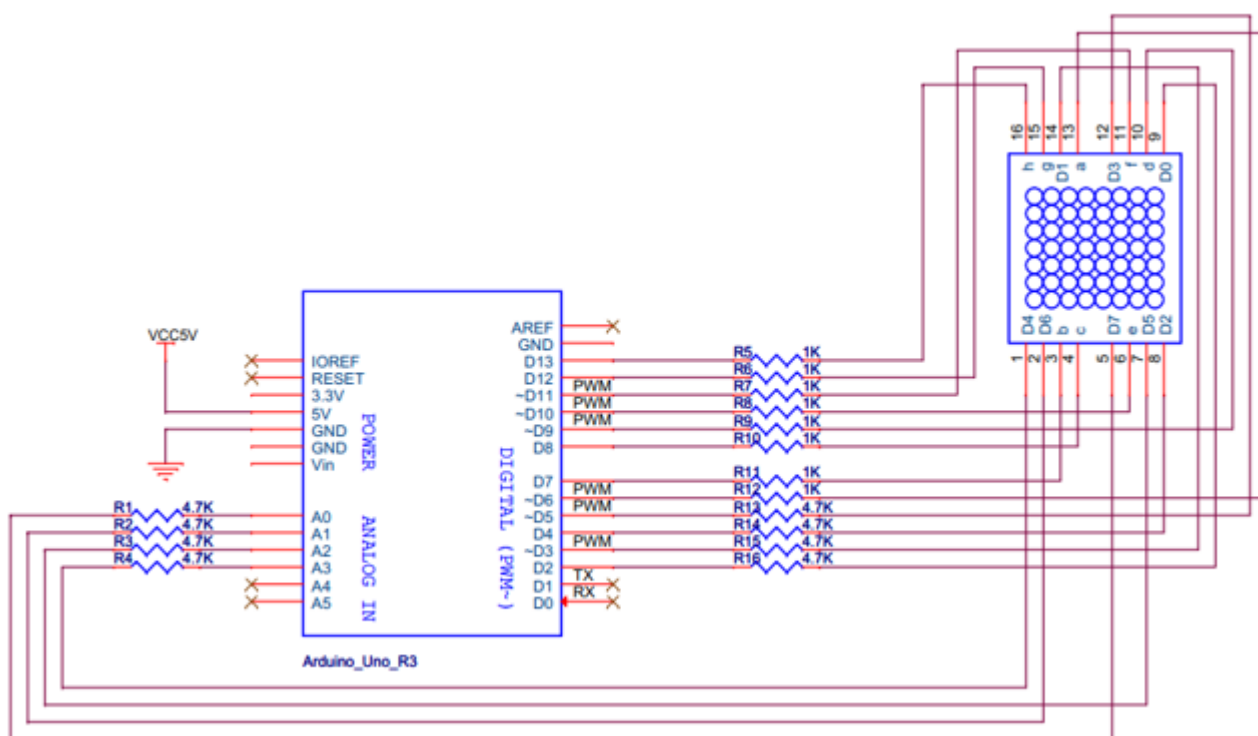
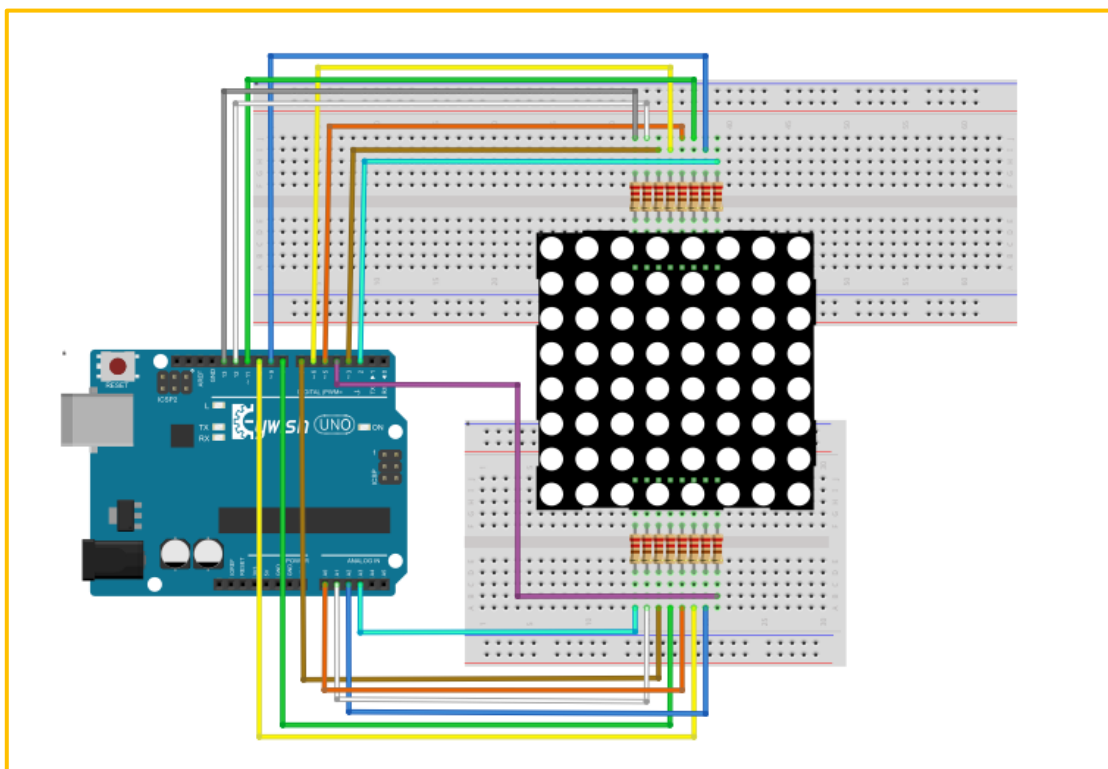


Схема подключения

Arduino Uno	8x8 Dot-matrix
7	7 (a)
6	6 (b)
5	4 (c)
10	2 (d)

11	1 (e)
8	9 (f)
9	10 (g)
4	5 (h)
GND	8 (gnd)
GND	3 (gnd)



Код

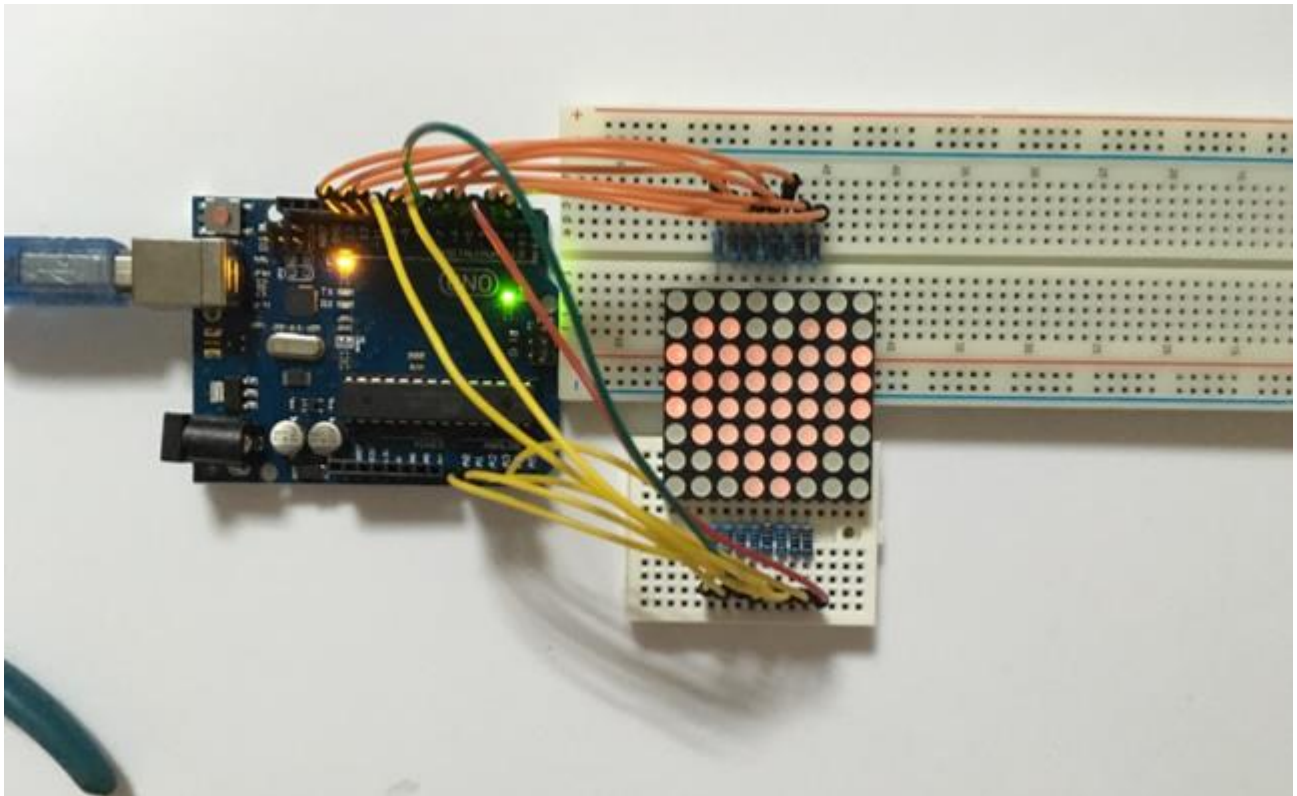
```
#define ROW_0 2 // 定义 ROW_ pin 9
#define ROW_1 3 // 定义 ROW_ pin 14
#define ROW_2 4 // 定义_ pin 8
#define ROW_3 5 // 定义_ pin 12
#define ROW_4 A3 // 定义_ pin 1
#define ROW_5 A2 // 定义_ pin 7
#define ROW_6 A1 // 定义_ pin 2
#define ROW_7 A0 // 定义_ pin 5
#define LED_A 6 // 定义 COL_ pin 13
#define LED_B 7 //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

● Результаты эксперимента



● Программа графического программирования Mixly

Программа для написания эксперимента 8×8 матрицы со Mixly показана на рисунке ниже:

```
8x8 lattice screen initialization ROW_0 2 ROW_1 3 ROW_2 4 ROW_3 5 ROW_4 A3 ROW_5 A2 ROW_6 A1 ROW_7 A0 LED_A 6 LED_B 7 LED_C 8 LED_D 9 LED_E 10 LED_F 11 LED_G 12 LED_H 13
Dot matrix display lightning
8x8 dot matrix screen continuously displays scan...
```

● Программа графического программирования MagicBlock

Программа для написания эксперимента 8×8 матрицы со MagicBlock показана на рисунке ниже:

```
setup
8x8 lattice screen initialization ROW_0 2 ROW_1 3 ROW_2 4 ROW_3 5 ROW_4 A3 ROW_5 A2 ROW_6 A1 ROW_7 A0 LED_A 6 LED_B 7 LED_C 8 LED_D 9 LED_E 10 LED_F 11 LED_G 12 LED_H 13
8x8 dot matrix display lightning
loop
8x8 dot matrix screen continuously scanning display
```