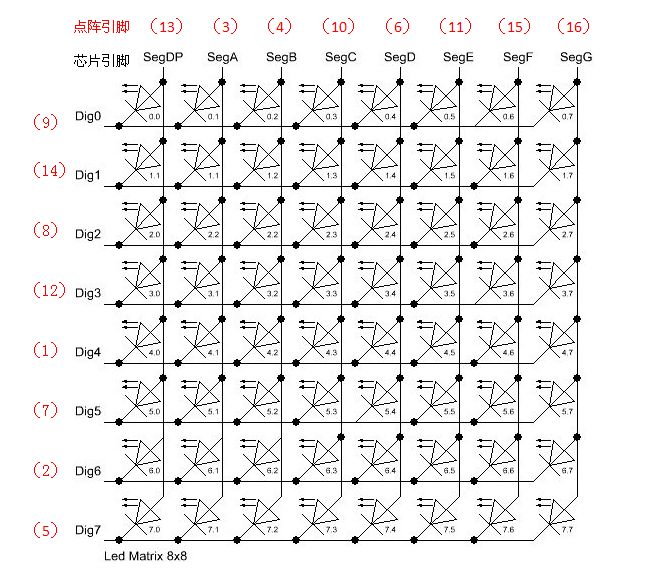
**8x8点阵**

1. **简介**

8x8点阵实际上是由64个led灯组成的点阵屏，屏幕上每一个点都是一个led，我们使用的是8\*8共阳极点阵，其中引脚16，15，11，6，10，4，3，13八个连接阳极(VCC)，其余引脚接阴极(GND)。



点阵控制基本原理：要点亮点阵上的某个点只要控制该点的阳极为高电平，阴极为低电平； 8\*8点阵每列二极管共用一个高电平，每行二极管共用用一个低电平。

1. MX7219

(1)直接使用8\*8点阵连接arduino的引脚非常不便利，需要占用16个引脚，极大地浪费arduino的引脚资源，而且连接线路也极为麻烦，MAX7219芯片提供了更好的解决方案。

(2)MAX7219是集成化串行输入/输出显示驱动器。串行指的是使用一条数据线，将按照二进制的形式一位一位地依次传输数据；

(3) MAX7219点阵模块引脚说明：

GND引脚：接GND

VCC引脚：接VCC（+5V）

DIN引脚：串行数据输入端口，用于发送数据

CS引脚：是片选信号，可以用于启动模块

CLK引脚：时钟序列输入端，发送时钟信号

注：课程中，我们将DIN引脚连接到arduino的A2引脚，CLK连接到arduino的A3引脚，CS连接到arduino的A4引脚

1. 基本语法
2. 导入点阵模块的库

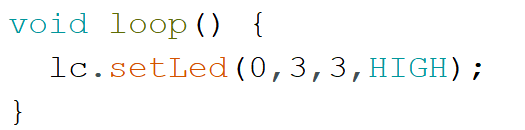
#include<LedControl.h> // LedControl库是基于MX7219来控制点阵模块的库，需要在arduino软件中下载

1. 在arduino软件的菜单栏中选择“项目”选项，然后把鼠标放在“加载库”上，右侧会弹出一个列表，水平移动鼠标，选择“管理库”
2. 在弹出的窗口中等待最下面的进度条走完，然后在上面的输入框中输入LedControl并按下键盘的“Enter”键
3. 待软件搜索完以后，找到LedControl的库，选择安装，等待下面的进度条结束就表示安装成功了

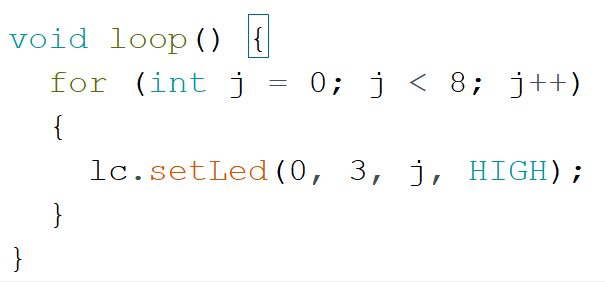
(2)LedControl lc = LedControl(A2,A3,A4,1);初始化点阵模块

示例中，A2表示点阵模块的DIN引脚连接arduino的A2引脚；A3表示点阵模块的CLK引脚连接到arduino的A3引脚；A4表示点阵模块的CS引脚连接到arduino的CS引脚；最后的数字1表示使用的点阵模块的个数，只能设置1~8的数字，也就表示最大能连续连接8个点阵模块

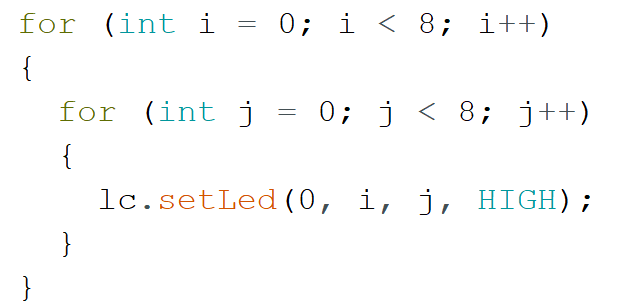
1. lc.shutdown(0,false);//启动点阵模块，其中0表示点阵模块的地址(编号);false为正常模式，如果改成true则为省电模式(关闭模式)
2. lc.setIntensity(0,8);//调节亮度，0表示点阵模块的地址；8表示亮度级别，级别可以设置0到15，数字越大，亮度越大
3. lc.clearDisplay(0);//清除显示，0表示点阵模块的地址
4. lc.setLed(0,i,j,HIGH);//0位点阵模块的地址，i和j表示点亮第i行第j列的led灯，HIGH表示高电平点亮，LOW表示低电平点亮，其中HIGH也可以用1代替，LOW可以用0代替
5. 课堂练习
6. 点亮第3行第3列的led灯



1. 点亮第3行所有的led灯



1. 点亮8行8列所有的led灯



1. 数组

数组可以保存同类型的多个变量，并可以按照序号访问数组中的数据。数组类似于python中的列表

一维数组：可以理解为用来保存一行的数据，定义如：int a[5]={1,2,3,4,5};

二维数组：可以存储一个矩阵，也可以看做是一维数组中的每个元素又是一个数组，例如：

int b[4][5]={{1,2,3,4,5},{6,7,8,9,3},{3,4,6,2,4},{3,1,2,6,5}}；//b[4][5]是一个二维数组，有4行5列，第一行为1，2，3，4，5。第一列为1，6，3，3。

也可以写成：

int b[4][5]={1,2,3,4,5,

6,7,8,9,3,

3,4,6,2,4,

3,1,2,6,5};

了解二维数组的结构，有助于我们控制点阵。

1. 编程练习

(1)在点阵屏上显示数字1

示例：

#include <LedControl.h>

LedControl lc = LedControl(A2, A3, A4, 1);

int a1[8][8] = {//1

0, 0, 0, 0, 0, 0, 0, 0,

0, 0, 0, 0, 1, 0, 0, 0,

0, 0, 0, 1, 1, 0, 0, 0,

0, 0, 0, 0, 1, 0, 0, 0,

0, 0, 0, 0, 1, 0, 0, 0,

0, 0, 0, 0, 1, 0, 0, 0,

0, 0, 0, 1, 1, 1, 0, 0,

0, 0, 0, 0, 0, 0, 0, 0,

};

void setup() {

lc.shutdown(0, false);

lc.setIntensity(0, 3);

lc.clearDisplay(0);

}

void loop() {

for (int i = 0; i <= 7; i++)

{

for (int j = 0; j <= 7; j++)

{

lc.setLed(0, i, j, a1[i][j]);

}

}

delay(100);

}