

用 **Arduino** 基础套件 DIY 出各种功能的电路

刘英杰

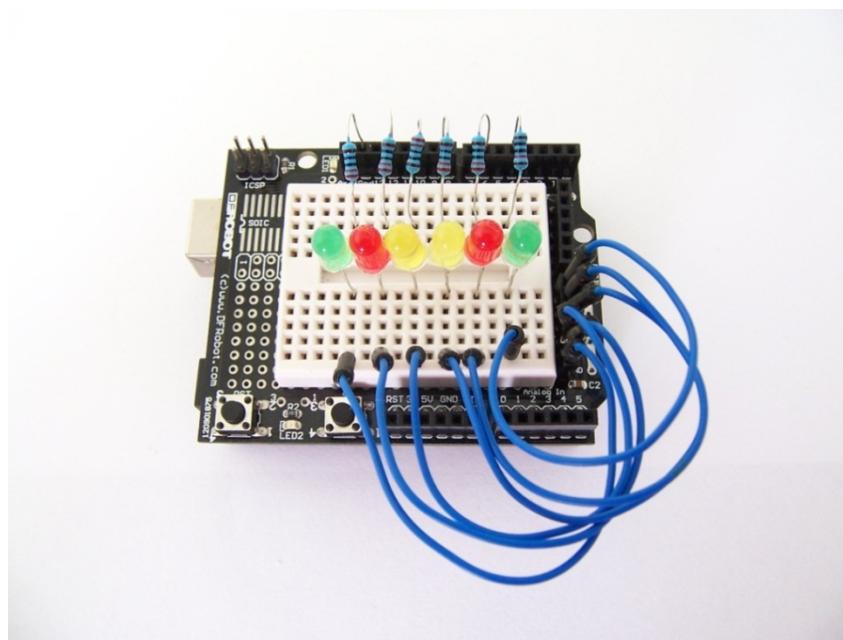
一直以来我就对电子方面很感兴趣，也想学点东西做出属于自己的作品，特别是在看到其他爱好者的作品之后，我的思想再一次波动，是该学点东西啦。经过向电子发烧友们请教后，发现有一种叫做 **Arduino** 的平台很对胃口。虽然 **Arduino** 的核心芯片也是单片机——**AVR** 单片机，但是 **Arduino** 的硬件平台是一块开放原始代码的 **Simple i/o** 平台，电路图是开源的，**Arduino** 的软件开发平台也是开源的，**Arduino** 的资源都可以免费下载。**Arduino** 的开发语言是建立在 **C/C++** 基础上的，其实也就是基础的 **C** 语言，**Arduino** 语言只不过把 **AVR** 单片机（微控制器）相关的一些参数设置都函数化，不用我们去了解他的底层，让我们不了解 **AVR** 单片机（微控制器）的朋友也能轻松上手。现在在全球有几万名开发人员，可以通过网上共享代码，分享智慧。在国外使用 **Arduino** 的用户，不仅是电子或者计算机相关专业的爱好者，其他行业例如：媒体艺术用来作出根据自己的想法可控灯光效果或者媒体互动的作品、园艺设计人员使用 **Arduino** 开发自己的智能浇灌系统、模型爱好者用在自己的无人飞机及其模型车上、无线电爱好者使用 **Arduino** 组成全自动莫尔斯电码解调器等，各个行业都在使用 **Arduino** 这一灵活方便的开发环境。

了解了 **Arduino** 平台后总想动手实践一下，于是乎就找了一款 **Arduino** 套件，套件里的器件简单易学，还配有相应的教程，很适合我们初识 **Arduino** 平台的人来练手。利用这些器件可以实现广告灯、模拟警笛声、掷骰子、抢答器、火焰报警、遥控等等有趣功能的电路。有了电路后我们就具备了硬件部分，软件部分可以在 **Arduino** 平台上编写对应功能的程序，将程序下载到控制板就可以实现对应功能啦。有很大的空间让我们来发挥哦。因为套件里备有面包板，所以整个制作过程也非常的简单，无须焊接，直接在面包板上插拔元器件即可。让您可以快速使用 **Arduino** 做出有趣的东西。在制作过程中一直有一种回到孩童时期玩积木的感觉，如果说孩童时期是用木头或塑料积木堆出各种形状的物体，那么现在就是用电子积木制作出具有各种趣味功能的电路。最重要的是，在体验乐趣的

同时，我们可以学到如何搭建具有某种功能的电路，如何根据硬件部分去编写对应的程序，硬件与软件之间的配合，了解了常用电子器件和传感器的使用方法。

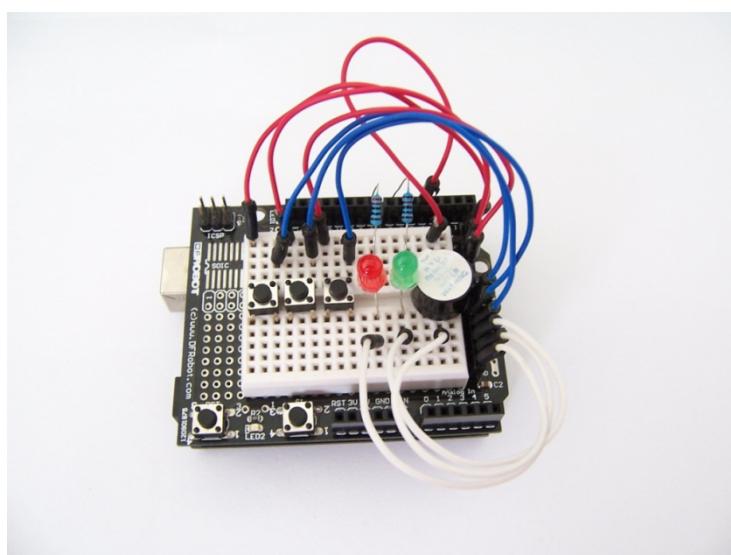
下面列举三个例子：

- 广告灯



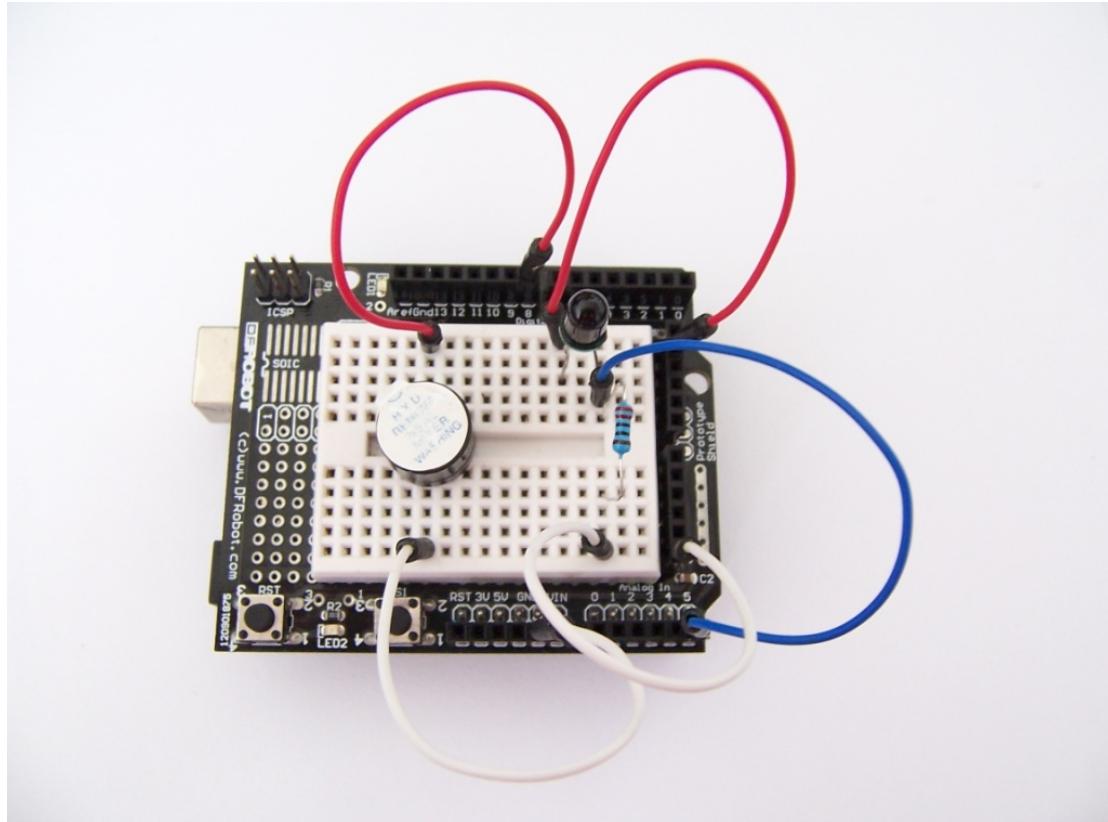
从套件配的文档中我们可以学到什么是 LED 灯、LED 的使用电路、多个 LED 灯如何一起配合使用、**for** 循环语句的使用。将广告灯程序下载到实验板后我们可以观察到，六个 led 不断的循环执行几种样式，非常漂亮。

- 抢答器



从这个例子中我们会学到：按键和蜂鸣器的结构，使用方法，如何用程序完成按键、led 灯、蜂鸣器之间的配合，如何用程序消除按键的抖动。抢答器程序功能：两名选手各有一个按键，当比赛开始后进行抢答，谁先按下按键对应的灯就会亮起来。裁判可根据亮灯情况提示参赛选手答题，本次结束后，裁判按下按键 3 清除现在亮灯情况（即将亮灯都熄灭）。每个按键按下时，蜂鸣器都会响，作为提示音。

- 火焰报警



从这个例子大家可以学到火焰传感器的工作原理，火焰传感器的使用电路。本程序可以模拟在有火焰时报警的情况，可以用打火机模拟火焰，当在火焰传感器附近打开打火机时，蜂鸣器发出响声报警，关掉火机，蜂鸣器停止发出声音。

下面就以掷骰子实验为例详细介绍 Arduino 套装使用过程。

所需材料：

1. Arduino 328控制板 1个

2. Prototype shield 扩展板 1个

3. 面包板 1个

4. LED 灯 红黄绿 各2个

5. 蜂鸣器 1个

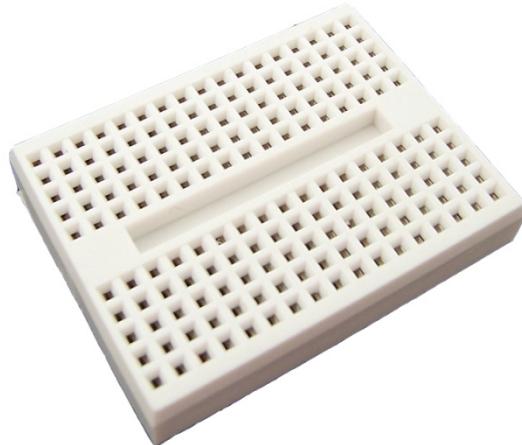
- 6. 6x5按键开关 4个
- 7. 数码管 1个
- 8. 倾斜开关 1个
- 9. 光敏电阻 1个
- 10. 红外接收三极管 1个
- 11. 温度传感器 1个
- 12. 红外接收头 1个
- 13. 电阻220欧、1K、10K 各3个
- 14. 6节5号电池盒 1个
- 15. USB 线 1条
- 16. 面包线 10条
- 17. mini 遥控器 1个



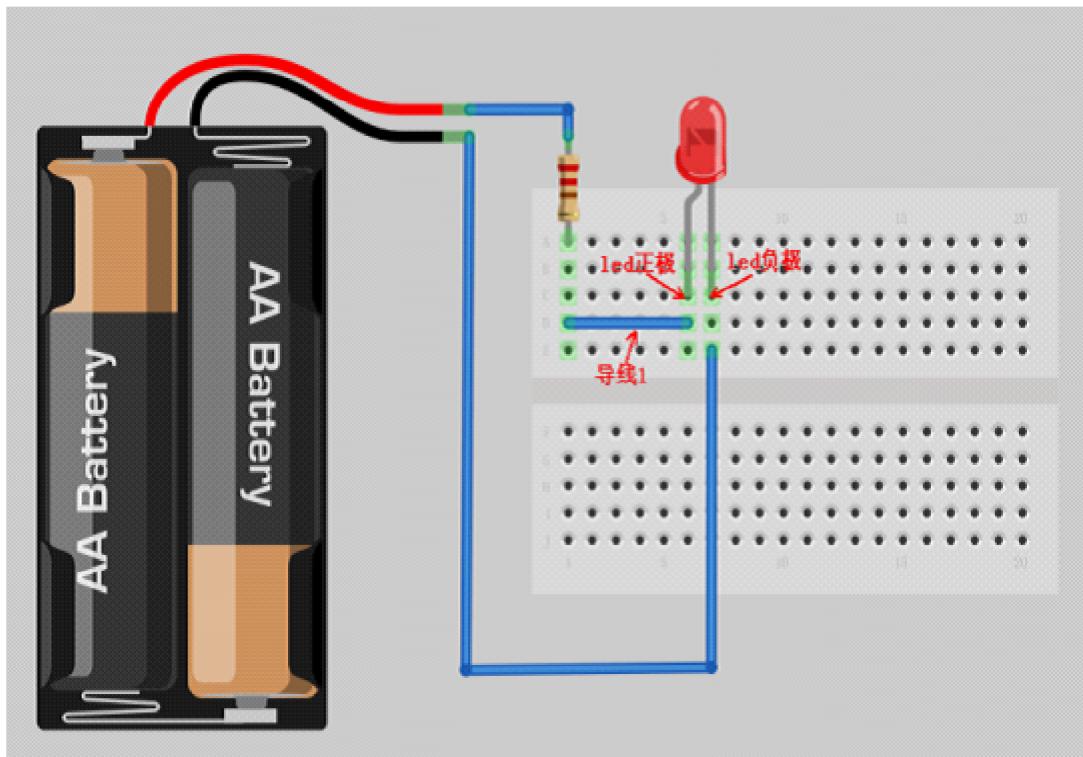
首先要向大家介绍一下 mini 面包板的使用方法：

优质迷你面包板具有**170**个插孔，此迷你面包板可以配合 **Arduino** 各种型号的 **ProtoShield**（原型扩展板），自带双面粘胶，可以粘贴到各种开发板、扩展板上，也可粘贴到各种轮式机器人或履带式机器人基板上实现个性化功能调试，体积小巧，仅有**45mm×35mm (1.8"×1.4")** 大小，是 **Arduino** 互动媒体爱好者、机器人发烧友、电子爱好者和电子工程师必备用品。

面包板（集成电路实验板）是电路实验中一种常用的具有多孔插座的插件板，在进行电路实验时，可以根据电路连接要求，在相应孔内插入电子元器件的引脚以及导线等，使其与孔内弹性接触簧片接触，由此连接成所需的实验电路。下图为本迷你面包板的示意图。



它一共具有**170**个插孔，**10**行**17**列。以中间的长槽为界分成上、下两部分，每一部分都是**5**行**17**列。从图中可以看到在每一部分中的每一列有**5**个插孔，这**5**个插孔的底部是一个金属簧片，因此插入这**5**个孔内的导线就被金属簧片连接在一起。例如下图：



电阻和红色发光二极管是连通的，因为导线1的一端插在了电阻所在的列，这列的5个孔是连通的，所以导线和电阻是通的，导线的另一端插在了发光二级管正极所在的列，所以最终电阻和发光二极管是连通的。将电源正负极接好后发光二极管就会被点亮。每一部分的每一列的五个插孔都是通过金属簧片相连的，而每一行的17个孔是不通的，所以横排上的器件要连通的话，需用导线连接。

注意：插入面包板上孔内引脚或导线铜芯直径为 0.4~0.6mm，即比大头针的直径略微细一点。元器件引脚或导线头要沿面包板的板面垂直方向插入方孔，应能感觉到有轻微、均匀的摩擦阻力，在面包板倒置时，元器件应能被簧片夹住而不脱落。面包板应该在通风、干燥处存放，特别要避免被电池漏出的电解液所腐蚀。要保持面包板清洁，焊接过的元器件不要插在面包板上。

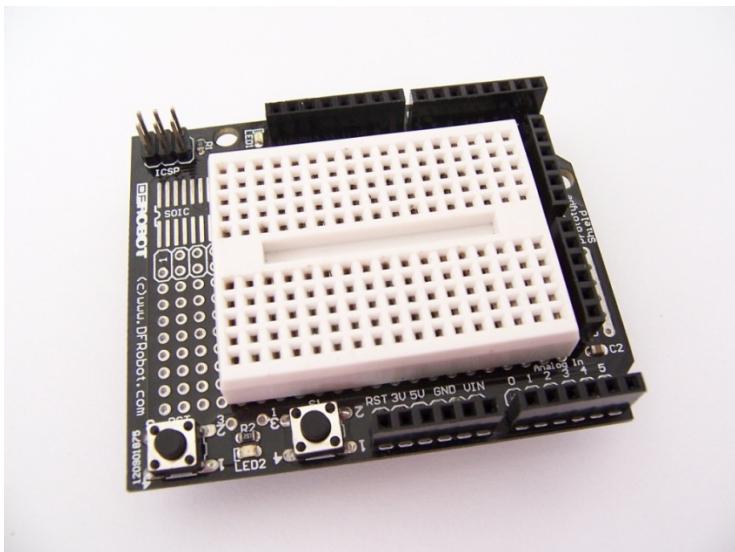
了解了以上内容后，接下来就跟着我一起动手制作吧：

一、硬件部分

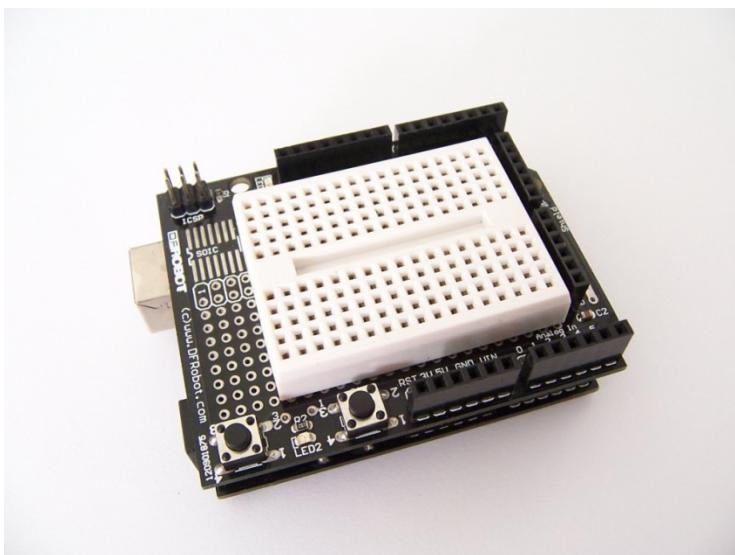
首先你需要准备所需要的材料 **Prototype shield** 扩展板1个、**mini 面包板**1个、**Arduino 328**控制板1个、按键1个、数码管1个、 220Ω 电阻8个、多彩面包板实验跳线若干条。

接下来，我们就开始动手啦。

- 1、将面包板背面的胶纸接下来，然后粘在 **Prototype shield** 扩展板上，完成后如下图所示。

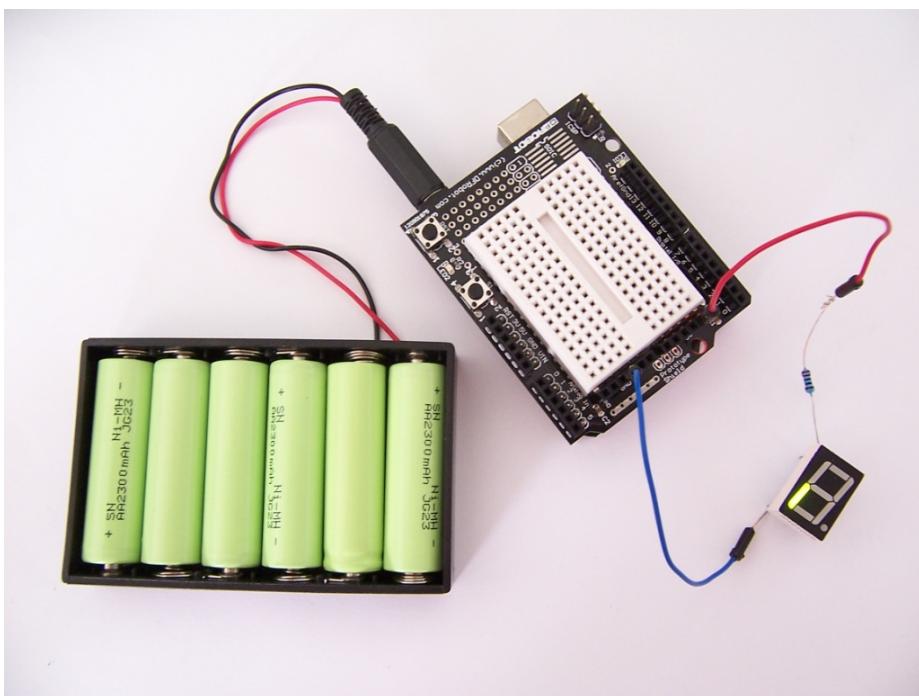


- 2、把贴有面包板的 **Prototype shield** 扩展板插在 **Arduino 328**控制板上，如下图。

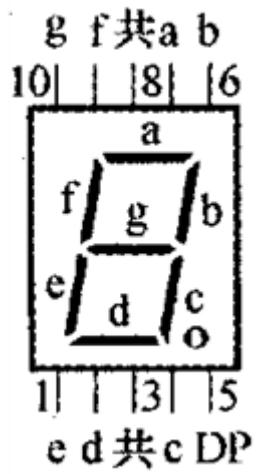


3、在用数码管之前，我们先要测试出数码管是共阴极还是共阳极的，还有数码管的每个字段是由哪一个引脚控制的。

首先我们将套件中的电池盒装满1.2V左右的5号电池，然后插在我们刚才已经插好的控制板上的电源接口，这时扩展板上面标有5V的插孔就有电了，拿一根面包线插在其中任意一个孔中，面包线的另一端串接一个220Ω的电阻，电阻的另一端我们把它称作A端（VCC），再拿一根面包线插在扩展板上标有GND的插孔中，这根面包线的另一端我们把它称作B端（GND），最后我们将A端和B端接在数码管的任意两个脚上，如下图：

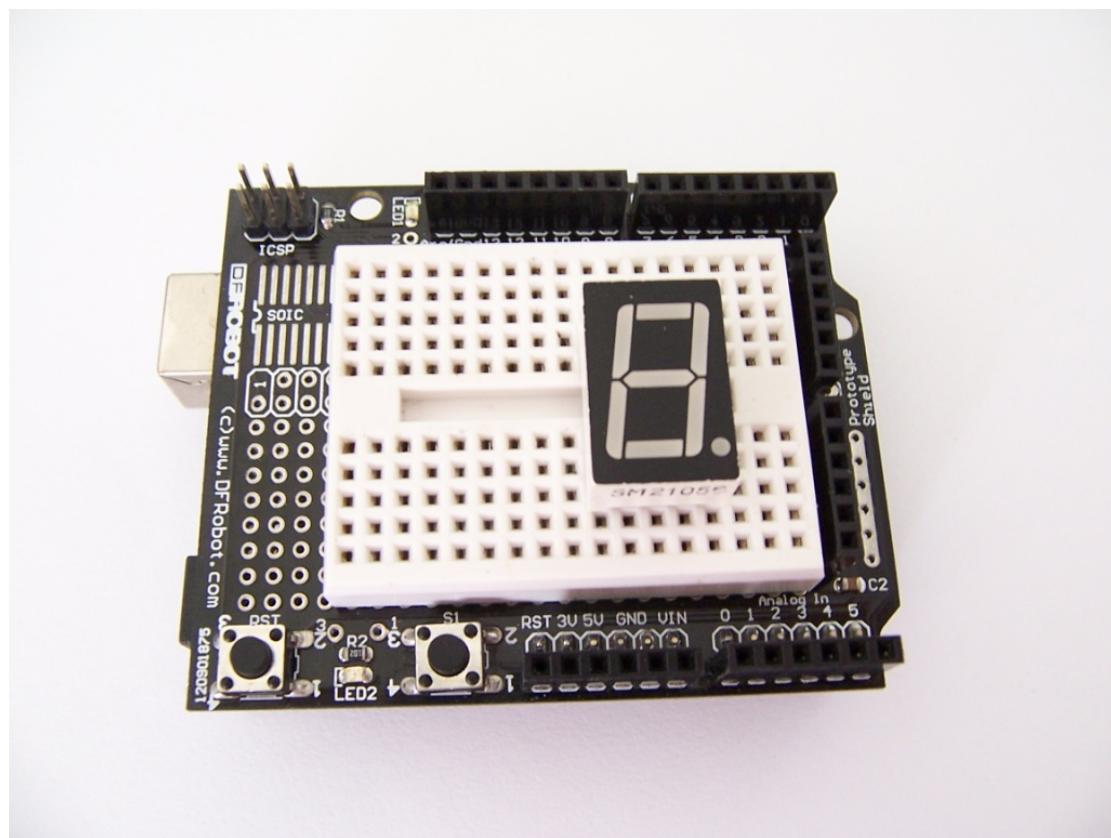


组合有很多，但总有一个字段会发光的，找到一个就够了，然后用B端（GND）不动，A端（VCC）逐个碰剩下的脚，如果有多个字段（一般是8个）发光，那它就是共阴的了，且B端就是公共极。相反用A端（VCC）不动，B端（GND）逐个碰剩下的脚，如果有多个字段（一般是8个）发光，那它就是共阳的了，且A端就是公共极。经过测试我用的数码管是共阳极的，且有两个共阳极引脚，将共阳极引脚记住。接下来我们测试一下每个字段是由哪一个引脚控制的，将刚才的A端与共阳极引脚连接，然后用B端接触剩余的字段引脚，测试后各引脚和字段对应关系如下图：



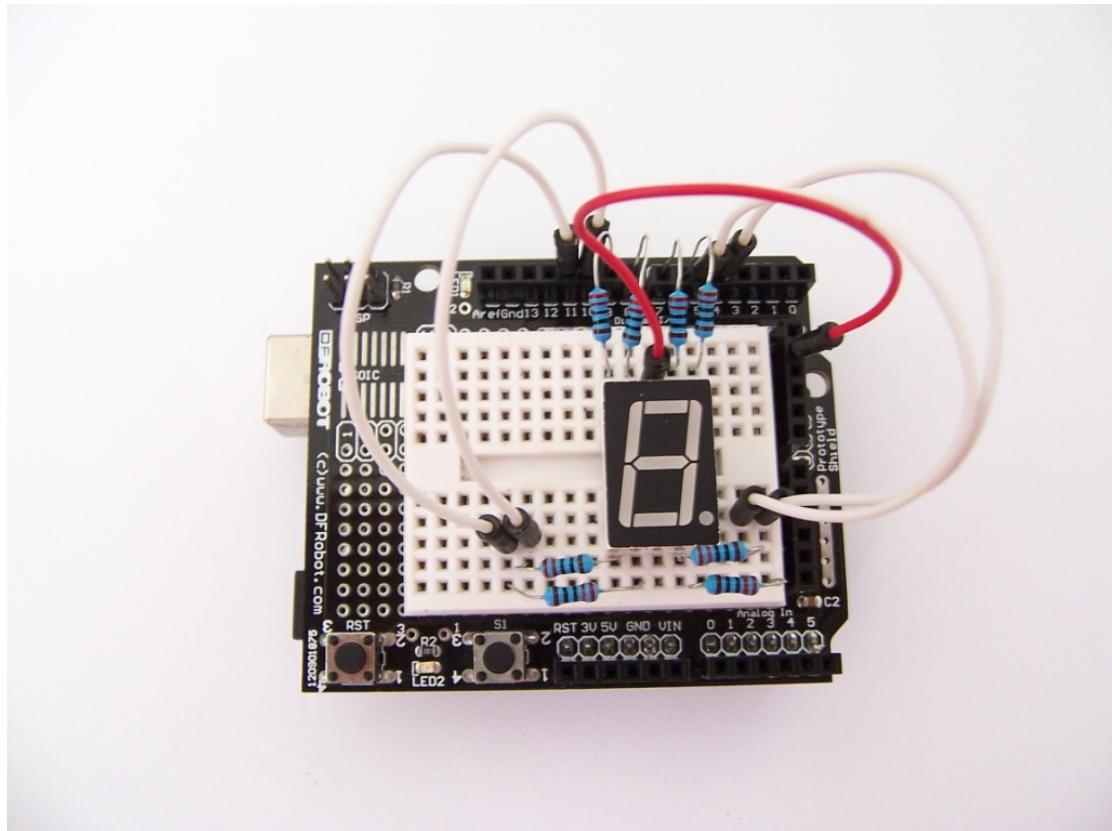
测试后将电源去掉，所有面包线面包线拔出，下面我们开始连接掷骰子功能电路啦。

4、将数码管插在面包板的合适位置

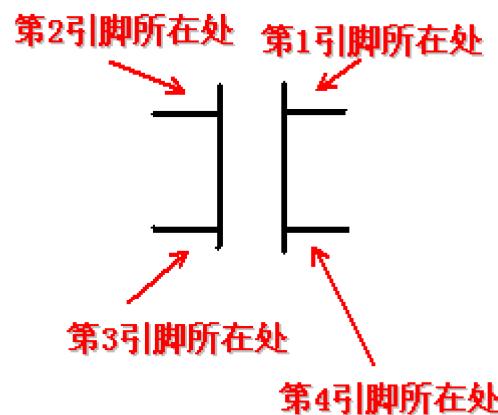


5、选择一根红色的面包线作为连接共阳极端的连线，以便于跟其它连线区分。将面包线的一端插在与数码管公共极引脚同列的插孔中，另一端插在**+5V**插孔中。接下来找出**220Ω**的电阻，将电阻的一端插在与 **a** 字段引脚同列的面包板插孔中，电阻的另一端插在扩展板的数字 **IO** 孔中任意一个即可，这里我插在了数字 **7** 口，剩下的引脚按照此方法插即可，这里 **a**、**b**、**c**、**d**、**e**、**f**、**g** 分别对应数字 **IO** 口的 **7**、

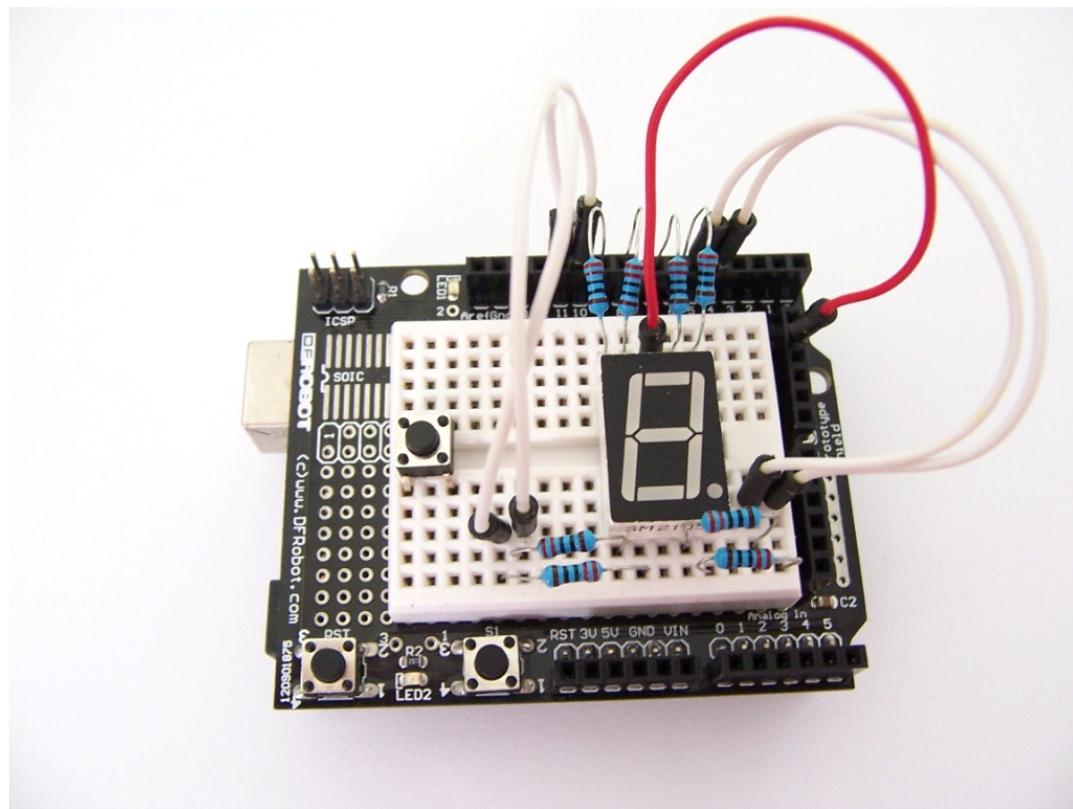
6、4、10、11、8、9，插在哪个数字 **IO** 口大家可以自己定只要记住就可以了，离数字 **IO** 口远的引脚可以先接电阻然后电阻的另一端通过面包线连接到数字 **IO** 中。这样数码管的部分就连接好了。如图：



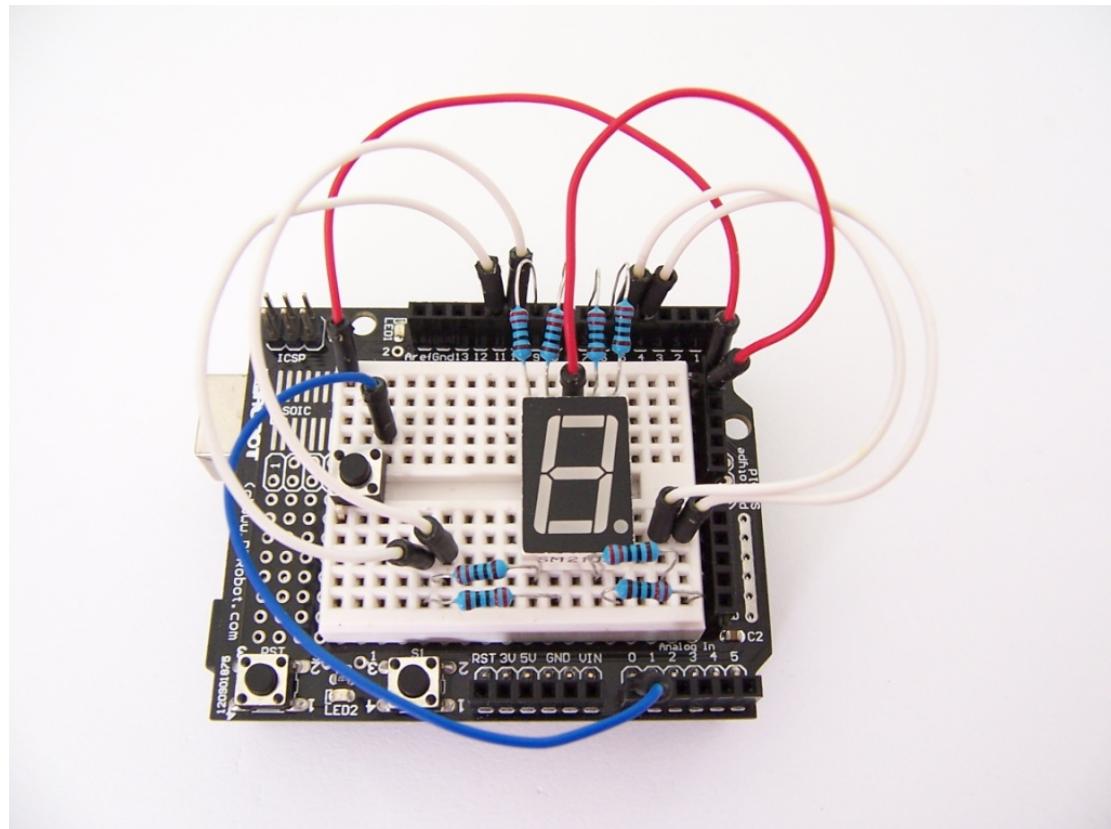
6、观察按键的背面，这时你可以看到按键背面如图：



这表示在没有按键按下时左侧两个引脚是导通的，右侧两个引脚也是导通的，而上侧和下侧两个引脚是没有导通的。当按键按下时上侧两个引脚导通，下侧也同样导通。按照上图的方向将按键插到面包板上，如图：



7、在按键上侧的左边引脚同列插上一根面包线，另一端插在+5V 插孔中，按键上侧的右边的引脚同列也插上一根面包线，面包线的另一端插在扩展板的模拟 0 口，如图：



哈哈，这样我们的硬件部分就连接好了。

二、软件部分

要实现：让数码管快速的显示数字 1~8。当按键按下时，模拟 0 口会读到电压值为 5V，只要电压值大于 2.5V 就停留在当前显示的数字；松开按键，电压值为 0，数字继续滚动。类似于，先摇动骰子，停止时骰子上面的数字即是我们摇出的点数。

数码管共有七段显示数字的段，还有一个显示小数点的段。当让数码管显示数字时，只要将相应的段点亮即可。例如：让数码管显示数字1，则将 b、c 段点亮即可。

程序代码如下：

```
//设置控制各段的数字 IO 脚

int a=7;
int b=6;
int c=5;
int d=11;
int e=10;
int f=8;
int g=9;
int dp=4;

//显示数字 1

void digital_1(void)
{
    unsigned char j;

    digitalWrite(c,LOW);//给数字 5 引脚低电平，点亮 c 段
    digitalWrite(b,LOW);//点亮 b 段
    for(j=7;j<=11;j++)//熄灭其余段
        digitalWrite(j,HIGH);
    digitalWrite(dp,HIGH);//熄灭小数点 DP 段
}

//显示数字 2

void digital_2(void)
```

```
{  
    unsigned char j;  
    digitalWrite(b,LOW);  
    digitalWrite(a,LOW);  
    for(j=9;j<=11;j++)  
        digitalWrite(j,LOW);  
    digitalWrite(dp,HIGH);  
    digitalWrite(c,HIGH);  
    digitalWrite(f,HIGH);  
}  
  
//显示数字 3  
  
void digital_3(void)  
{  
    unsigned char j;  
    digitalWrite(g,LOW);  
    digitalWrite(d,LOW);  
    for(j=5;j<=7;j++)  
        digitalWrite(j,LOW);  
    digitalWrite(dp,HIGH);  
    digitalWrite(f,HIGH);  
    digitalWrite(e,HIGH);  
}
```

```
//显示数字 4

void digital_4(void)

{
    digitalWrite(c,LOW);
    digitalWrite(b,LOW);
    digitalWrite(f,LOW);
    digitalWrite(g,LOW);
    digitalWrite(dp,HIGH);
    digitalWrite(a,HIGH);
    digitalWrite(e,HIGH);
    digitalWrite(d,HIGH);

}

//显示数字 5

void digital_5(void)

{
    unsigned char j;
    for(j=7;j<=9;j++)
        digitalWrite(j,LOW);
    digitalWrite(c,LOW);
    digitalWrite(d,LOW);
    digitalWrite(dp,HIGH);
    digitalWrite(b,HIGH);
```

```
digitalWrite(e,HIGH);

}

//显示数字 6

void digital_6(void)

{

unsigned char j;

for(j=7;j<=11;j++)

digitalWrite(j,LOW);

digitalWrite(c,LOW);

digitalWrite(dp,HIGH);

digitalWrite(b,HIGH);

}

//显示数字 7

void digital_7(void)

{

unsigned char j;

for(j=5;j<=7;j++)

digitalWrite(j,LOW);

digitalWrite(dp,HIGH);

for(j=8;j<=11;j++)

digitalWrite(j,HIGH);

}
```

```
//显示数字 8

void digital_8(void)

{

    unsigned char j;

    for(j=5;j<=11;j++)

        digitalWrite(j,LOW);

    digitalWrite(dp,HIGH);

}

void setup()

{

    int i;

    for(i=4;i<=11;i++)

    {

        pinMode(i,OUTPUT);//设置 4~11 口味输出模式

    }

}

void loop()

{

    while(1)

    {

        digital_1();//显示数字 1
    }
}
```

```
while(analogRead(0)>512);//如果读到模拟 0 口的值为 0 则说明有按键  
按下  
delay(200);//延时 200ms  
  
digital_2();  
  
while(analogRead(0)>512);  
  
delay(200);  
  
digital_3();  
  
while(analogRead(0)>512);  
  
delay(200);  
  
digital_4();  
  
while(analogRead(0)>512);  
  
delay(200);  
  
digital_5();  
  
while(analogRead(0)>512);  
  
delay(200);  
  
digital_6();  
  
while(analogRead(0)>512);  
  
delay(200);  
  
digital_7();  
  
while(analogRead(0)>512);  
  
delay(200);  
  
digital_8();  
  
while(analogRead(0)>512);
```

```
    delay(200);

}

}
```

将程序下载到实验板后我们可以观察到，数码管不断的循环显示数字 1~8，当按下按键后，数码管显示当前数字，直至松开按键。

Arduino 基础套装硬件资源丰富，学习资料也非常的丰富有趣。附带的十节课程的编排完全是从初学者的角度考虑，每一节实验都配有图文结合的实验说明文档和非常有趣的例子程序。而且每一节实验除了文档上讲的方法外，还有很大可供学习者发挥的空间。如果您是刚刚接触电子的初级爱好者，那赶紧行动吧，用 Arduino 套装来丰富您的电子知识，DIY 出你的第一件作品。如果您是对电子感兴趣的学生，那您可以用 Arduino 套装做实验，来学习电子知识、开发您电子方面的潜能。如果您是传统的电子爱好者，您可以利用 Arduino 套装来了解 Arduino 这个开源平台，丰富您的电子知识。以上就是我的心得了，如有不足请大家多多指教，让我们共同进步。