

更多原创作品尽在电路城: <http://www.cirmall.com/>

Arduino 入门教程(3)—做一个 S.O.S 求救

简单回顾下 Lesson 2 的内容：

- 如何使用面包板
- 电阻的作用
- LED 相关知识

STEP 1 : 还原 Lesson 2 的硬件电路

这次将继续使用 Lesson 2 的搭建的电路，但我们这里将改变一下代码，就能让 LED 变为 S.O.S 求救信号了。这是国际莫尔斯码求救信号。莫尔斯码是一种字符编码，英文的每个字母，都是由横杠和点不同的组合而成。这样的好处是，使用简单的两种状态，就能来传递所有的字母和数字，非常的简便！可以通过 LED 开关两种状态来拼出一个个字母。长闪烁和短闪烁来表示点和横杠。就拼写 S.O.S 这三个字母。

通过查阅莫尔斯码表，可以知道，**字母“S”用三个点表示，我们这里用短闪烁替代，字母“O”则用三个横杠表示，用长闪烁替代。**

有 Lesson 1 和 Lesson 2 的基础，不难理解下面这段代码吧！

```
1. int ledPin = 10;
2. void setup() {
3.     pinMode(ledPin, OUTPUT);
4. }
5. void loop() {
6.     // 三个快闪烁来表示字母“s”
7.     digitalWrite(ledPin,HIGH);
8.     delay(150);
9.     digitalWrite(ledPin,LOW);
10.    delay(100);
```

更多原创作品尽在电路城: <http://www.cirmall.com/>

```
11.
12.     digitalWrite(ledPin,HIGH);
13.     delay(150);
14.     digitalWrite(ledPin,LOW);
15.     delay(100);
16.
17.     digitalWrite(ledPin,HIGH);
18.     delay(150);
19.     digitalWrite(ledPin,LOW);
20.     delay(100);
21.
22.     //三个短闪烁来表示字母“O”
23.     digitalWrite(ledPin,HIGH);
24.     delay(400);
25.     digitalWrite(ledPin,LOW);
26.     delay(100);
27.
28.     digitalWrite(ledPin,HIGH);
29.     delay(400);
30.     digitalWrite(ledPin,LOW);
31.     delay(100);
32.
33.     digitalWrite(ledPin,HIGH);
34.     delay(400);
35.     digitalWrite(ledPin,LOW);
36.     delay(100);
37.
38.     //再用三个快闪烁来表示字母“S”
39.     digitalWrite(ledPin,HIGH);
40.     delay(150);
41.     digitalWrite(ledPin,LOW);
42.     delay(100);
43.
44.     digitalWrite(ledPin,HIGH);
45.     delay(150);
46.     digitalWrite(ledPin,LOW);
```

更多原创作品尽在电路城: <http://www.cirmall.com/>

```
47.     delay(100);
48.
49.     digitalWrite(ledPin,HIGH);
50.     delay(150);
51.     digitalWrite(ledPin,LOW);
52.     delay(100);
53. }
```

STEP 2 : 输入代码

上面的写法固然正确，可是是不是觉得有点繁琐呢？如果有个 100 个，难不成还重复 100 遍吗？有没有更好的书写程序的方法呢？想必发明编程的人也考虑到这个问题了，所以有了我们更好的一种写法。我们先来看一下下面这段代码。

```
1.  int ledPin = 10;
2.  void setup() {
3.      pinMode(ledPin, OUTPUT);
4.  }
5.  void loop() {
6.      // 三个快闪烁来表示字母“s”
7.      for(int x=0;x<3;x++){
8.          digitalWrite(ledPin,HIGH);          //设置 LED 为开
9.          delay(150);                          //延时 150 毫秒
10.         digitalWrite(ledPin,LOW);             //设置 LED 为关
11.         delay(100);                          //延时 100 毫秒
12.     }
13.
14.     //100 毫秒延时产生字母之间的间隙
15.     delay(100);
16.
17.     //三个短闪烁来表示字母“o”
18.     for(int x=0;x<3;x++){
19.         digitalWrite(ledPin,HIGH);            //设置 LED 为开
20.         delay(400);                          //延时 400 毫秒
21.         digitalWrite(ledPin,LOW);             //设置 LED 为关
22.         delay(100);                          //延时 100 毫秒
23.     }
```

更多原创作品尽在电路城: <http://www.cirmall.com/>

```
24.
25.      //100 毫秒延时产生字母之间的间隙
26.      delay(100);
27.
28.      // 再用三个快闪烁来表示字母"s"
29.      for(int x=0;x<3;x++){
30.          digitalWrite(ledPin,HIGH);          //设置 LED 为开
31.          delay(150);                          //延时 150 毫秒
32.          digitalWrite(ledPin,LOW);            //设置 LED 为关
33.          delay(100);                          //延时 100 毫秒
34.      }
35.
36.      // 在重复 S.O.S 信号前等待 5 秒
37.      delay(5000);
38. }
```

在输入代码的时候，注意保持代码的一个层次感，除了美观外，也便于你日后检查代码。确认正确后，下载代码到 Arduino 中，如果一切顺利的话，我们将看到 LED 闪烁出莫尔斯码 S.O.S 信号，等待 5 秒。重复闪烁。给 Arduino 外接电池，整个装到防水的盒子里，就可以用来发 S.O.S 信号了。S.O.S 通常用于航海或者登山。我们接着来分析下代码。

代码回顾

代码的第一部分与 Lesson 2 是完全一样的。设置数字引脚 10 的模式为输出模式。在主函数 loop()中，你可以看到与上一个课中类似的语句用来控制 LED 的开和关，并保持一段时间。然而，这次不同的是，主函数包含了三个独立的代码段。

第一段代码是输出是三个点：

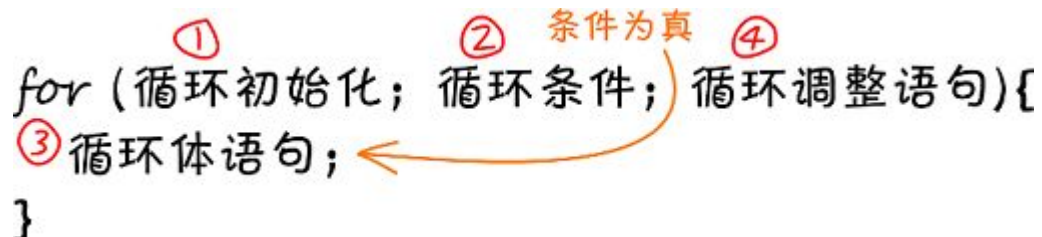
```
1.      for(int x=0;x<3;x++){
2.          digitalWrite(ledPin,HIGH);          //设置 LED 为开
3.          delay(150);                          //延时 150 毫秒
```

更多原创作品尽在电路城: <http://www.cirmall.com/>

```
4.    digitalWrite(ledPin,LOW);           //设置 LED 为关
5.    delay(100);                          //延时 100 毫秒
6.    }
```

LED 开关的语句是包含在一对花括号内的,因此为一组代码段。**必须说明的,花括号必须成对出现,如有遗留编译器编译时将不通过。**我是这么干的,在开始写花括号的时候,就先把 “{” “}” 都写上,之后再在两个括号之间输入代码,这样就不会出现写到最后括号对应不上情况。

当程序运行后,我们可以看到,灯闪了 3 次而不是只闪了 1 次。产生这样效果的是因为使用了 for 循环。for 语句通常在程序中用作循环使用。


for (循环初始化; 循环条件; 循环调整语句){
③ 循环体语句;
}

for 循环顺序如下:

第一轮: 1 --> 2 --> 3 --> 4

第二轮: 2 --> 3 --> 4

...

直到 2 不成立, for 循环结束。

来看下我们程序中的 for 循环:

```
1.    for(int x=0;x<3;x++){
2.    .....
3.    }
```

第一步: x 开始为 0;

第二步: 判断 x 是否小于 3。

第三步: 判断第二步成立, for 循环中执行 LED 开与关。

更多原创作品尽在电路城：<http://www.cirmall.com/>

第四步：x 自加，变为 2。

这里有个 x++，这是什么意思呢？

x++这句话表示把 x 的值增加 1，等同于写成 x=x+1，也就是把 x 当前的值变为 x+1。

第五步：回到第二步，此时 x=2，判断是否小于 3。

第六步：重复第三步。

.....

直到 x 循环到 3 时，判断 $x < 3$ 不成立，自动跳出 for 循环，程序继续往下走。

现在知道 for 循环是如何运作吧！我们代码中有 3 个 for 循环：第一个 for 循环 3 次，长闪烁 3 次，代表字母 “S”。第二个 for 循环同样循环 3 次，短闪烁 3 次，也就是字母 “0”。第三个 for 循环又来输出个 “S”。

在每 for 循环之间有个小延时 100 毫秒，是 S.O.S 字母之间有个清晰的停顿说明。

最后，在回到主函数 loop 重新执行一遍之前，有个相对较长的延时，为 5 秒。

好了，我们 S.O.S 信号源项目就算告一个段落了。有所收获吗？

可以做一些其他的事

有了一定的基础，现在做个课后习题吧~做个交通信号灯，下图是整个一个运行过程，虚线框的是程序循环的部分。

提示：上面我们是只点亮的 LED 灯，现在需要点亮三个 LED 灯。电路连接的原理是和一个灯相同，程序中需要改变的用三个数字口来分别控制 3 个 LED 灯。自己动手试一下吧！

更多原创作品尽在电路城: <http://www.cirmall.com/>

