

让呼吸灯的效果更加真实

谢作如 浙江省温州中学

李琦 浙江省诸暨市牌头中学

学科关键词: 数学、物理、技术

呼吸灯,顾名思义,就是灯光在单片机的控制下完成由亮到暗的逐渐变化,感觉像是在呼吸。呼吸灯广泛应用于手机上,并曾经成为各大品牌手机的卖点之一。如果你的手机里有未处理的通知,如未接来电、未查收的短信等,呼吸灯就会不断由暗到亮变化,像呼吸一样有节奏,起到提醒的作用。

学习Arduino,呼吸灯是一个典型的案例。笔者编写的《Arduino创意机器人》(人民邮电出版社出版)一书中就有“呼吸灯”一课,位于第一章的LED专题。通过研究呼吸灯效果,来学习Arduino的PWM输出,并且学习Arduino的循环语句和全局变量。但从效果看,这还未达到理想的效果,因为灯光强弱的过渡不够柔和。

● 为什么呼吸灯的效果不够真实

《Arduino创意机器人》中是使用两个循环来实现0到255和255到0的数值变换。程序是用ArduBlock编写,参考代码如图1所示。

在这两个循环中,变量K的值是递增到255,然后再降到0。如何让呼吸接近自然、真实?单纯看代码自然看不出什么。我们可以试着

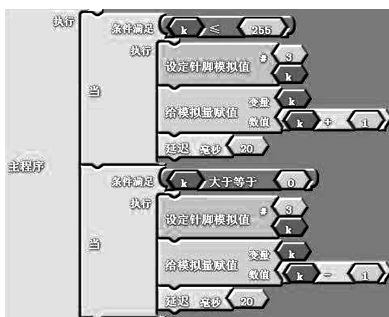


图1

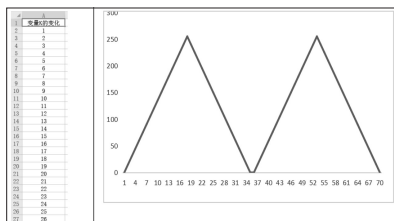


图2

用Excel输出从0到255再到0的一组数字,再用图表画出变量K的变化,如图2所示。

这时,我们就能明显看出线条是笔直的,成尖峰状。这样的数据变化,当然很难体现出“呼吸”这一张一弛的自然变化。那么,如何引入一个数学表达式,让图2的线条变得圆滑起来呢?

● 正弦函数的作用分析

在初中数学中,三角函数是一个重要的知识点,而正弦曲线就是一条圆滑的波浪线。嵌入式系统中许多应用都是对正弦函数进行采样,得到按正弦规则变化的数组。

举个比较简单的例子,如呼吸灯、警报声等。

关于正弦函数,其定义如下:在直角坐标系中,给定单位圆,对任意角 α ,使角 α 的顶点与原点重合,始边与x轴非负半轴重合,终边与单位圆交于点P(u, v),那么点P的纵坐标v叫作角 α 的正弦函数,记作 $v=\sin(\alpha)$ 。通常,我们用x表示自变量,即x表示角的大小,用y表示函数值,这样我们就定义了任意角的三角函数 $y=\sin(x)$,它的定义域为全体实数,值域为 $[-1,1]$ 。

这段文字看起来有些专业,其实我们只需要了解一句话:在 $y=\sin(x)$ 的表达式中,不管x的值是什么,y的值总在-1到1之间,而且是小数。那么如何生成这组数据,并且数据呈现的线条是怎样的,我们还是用Excel软件来研究一下。

笔者先在Excel中产生0~360的一组角度数字,然后用RADIANS函数转换为弧度,再用Sin函数输出数据。因为Arduino的PWM在0和255之间,就乘以255,这样得到了-255到255之间的数字,如下页表1所示。

同样用图表来绘制(如下页图3)。看起来是不是很平滑啊?

表1 计算Sin函数

角度	弧度	转换为PWM
0	0	0
1	0.017453293	4
2	0.034906585	8
3	0.052359878	13
4	0.06981317	17
5	0.087266463	22
6	0.104719755	26

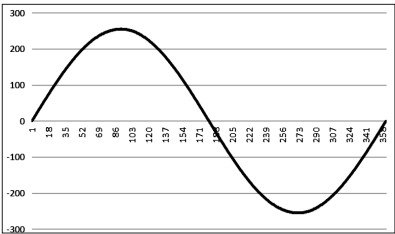


图3

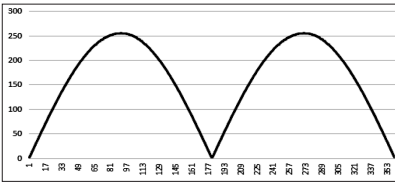


图4

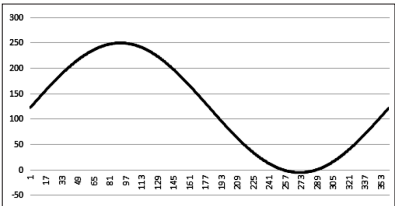


图5

```
int i;
void setup() {
  pinMode(6, OUTPUT); // 要选择支持 PWM 输出的端口，如 3、5、6、
  Serial.begin(9600);
}
void loop() {
  int pwm;
  pwm = (int(sin(radians(i))*255)/2)+128;
  analogWrite(6, pwm);
  Serial.println(pwm); // 在串口输出数据，可以帮助调试。
  i++;
  delay(10); // 适当修改延时的数值，可以调节呼吸的频率。
}
```

图6

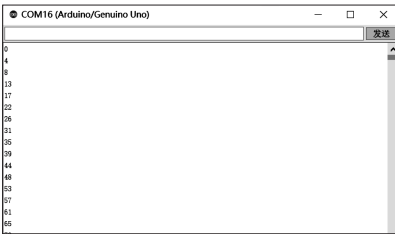


图7

至于如何将-255到255之间的数转化为0~255的数，有几种方案：

①取绝对值法。直接取绝对值，即使用ABS函数。这样一来，绘出来的线条就变成了有圆顶的山丘状，如图4所示。

②数据映射法。将-255到255之间的数映射为0~255之间。采用的办法也不复杂，除以2后，加上128即可，生成的数值如图5所示。Arduino编程就更简单了，可以直接调用map函数。

● 代码编写和效果实现

研究了正弦函数后，接下来就是编写代码了。Arduino中提供了radians、sin、int、abs、map等函数，因此写这个表达式非常容易。这些函数的简介如下。

radians：使用范例为radians(value)。将度数value转换为弧度，如果不太在乎误差，可以用“乘以3.14159再除以180”来替代。

sin：使用范例为sin(value)。返回角度value (radians形式) 的三角函数sine值，数据类型为double。

int：使用范例为int(value)。返回向下取整为最接近value的整数。

abs：使用范例为abs(value)。返回value的绝对值，可以将负数转正数。

map：使用范例为map(value, fromLow、fromHigh、toLow、toHigh)。按照fromLow与fromHigh范围，对等转换value至toLow与toHigh范围。

表2 三种方式的比较

方式	代码	效果
1	pwm=abs(int(sin(radians(i))*255));	有跳动感
2	pwm=(int(sin(radians(i))*255)/2)+128;	自然真实
3	pwm=map(int(sin(radians(i))*255),-255,255,0,255);	

具体的参考代码如图6所示。

经过比较，直接采用绝对值形式的效果不如第二种数据映射的方式，在变暗的时候有跳动感。而方式2和方式3的表达式基本上是等价的，如表2所示。

需要强调的是，调试程序要善于利用串口输出功能，即Serial.print。这样可以帮助我们了解变量的数据变化，如图7所示。

应用了正弦函数后，Arduino做出来的呼吸灯就很有“呼吸”的感觉了。当然，实现呼吸效果除了用单片机编程来实现外，也可以用模拟电路来实现。希望这个案例能让大家更加深入了解数学知识在编程中的应用。

如果对相关内容感兴趣，请关注主持人博客。

