

# 无线电

RADIO Magazine  
定价: 8.00 元

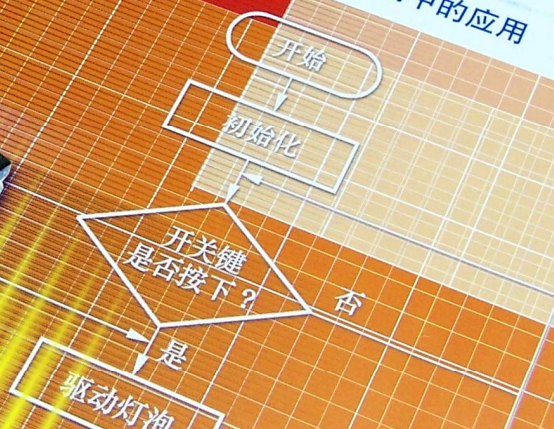
1955 年创刊  
2008  
www.radio.com.cn  
国内邮发代号: 2-75  
国外代号: M108

## 单片机编程魔法学校

★建立编程平台 ★实际动手改写程序 ★单片机在音响中的应用



ISSN 0512-4174  
9 770512 417009



## 五花八门，想变就变

### 实际动手改写程序

文/杜 洋

从一开始我就劝大家用 C 语言写程序，本篇文章也主要是在讲 C 语言编程。人们常说：杜洋卖瓜，自卖自夸。可多算是小贩，我只是希望大家可更高效地学习单片机。话不服，现在我就证明给你看，让你心口服、全家福。

货比两家

(4) 也是一种伪指令，这是 FLASH 的什么位置，想详细了

```
LED EQU P1.0
ORG 00H
LJMP ST
```

[程序 1]

;你的 P1



## 第三章 《五花八门，想变就变》

### 货比两家

从一开始我就劝大家用 C 语言写程序，本篇文章也主要是在讲 C 语言编程。人们常说：杜洋卖瓜，自卖自夸。可是 C 语言并不是我发明的，我顶多算是小贩，我只是希望大家可以更高效的学习单片机。话不服人，现在我就证明给你看，让你心服、口服、全家福。

#### [程序 1]

```
LED EQU P1.0           ;你的 P1.0 引脚上接了一个发光二极管      (1)

ORG 00H                 ;                                          (2)
LJMP START              ;跳到 START 标号处开始运行                (3)
ORG 30H                 ;                                          (4)

START:                  ;程序标号                                  (5)
    CLR LED             ;点亮发光二极管                            (6)
    LCALL DELAY          ;亮 1 秒钟（延时 1 秒钟）                (7)
    SETB LED            ;关掉发光二极管                            (8)
    LCALL DELAY          ;关 1 秒钟                                (9)
    LJMP START           ;跳到 START 标号处执行                    (10)

DELAY: MOV R5,#8         ;延时子程序，延时 1 秒钟                (11)
D0: MOV R7,#250          ;
D1: MOV R6,#250          ;
D2: DJNZ R6,D2           ;
    DJNZ R7,D1           ;
    DJNZ R5,D0           ;
RET                      ;子程序结束                              (12)

END                      ;全部程序到此结束                        (13)
```

请各位先看一下[程序 1]，不管你是否相信，这都是一个完整的、可以直接应用的汇编程序。程序可以实现让接在 P1.0 接口的发光二极管闪烁，闪烁频率是 1 秒钟。现在假想你就是一块单片机，看看你怎么按顺序执行这些程序吧，当你看不懂时请回到这里，我会有所解释。现在开始了，(1) 是一种伪指令，意思是定义单片机上的 P1.0 接口用“LED”表示，这样我们可以很方便地修改接口，当然你也可以直接把 P1.0 写在程序里，如果再次修改时你不怕麻烦的话。(2) 和 (4) 也是一种伪指令，这是指定程序写在单片机内容 FLASH 的什么位置，想详细了解什么是伪指令的话请自己去找答案，这里不用管它，当它们不存在是不会影响程序思路的。(3) 的意思是让程序跳到 START 标号处运行，你是单片机，你要找到 START 在什么地方。没错，就是 (5) 了，“标号”不是程序，它就像书的页码一样不算作文章的内容却必不可少，让你翻到哪页你就翻到哪页继续运行程序。下一句是 (6)，指令的含意是将 LED（也就是 P1.0 接口）变成低电平，回头看一下实验板的电路吧，P1.0 接口变成低电平后发光二极管会点亮还

是熄灭呢？回答熄灭的朋友请回家补习一下电路基础课。下一句是（7），意思是调用标号是“DELAY”的子程序。子程序的概念在令后是会经常用到的，我们在编程的时候一定会发现有一些重复内容，比如延时 1 秒钟在我们的闪烁程序中会用到两次，当然你也可以写两个完全一样的程序内容按顺序执行，不过当需要重复使用 N 次延时程序且 N 大于 100 时看你傻不傻眼。子程序就是将重复使用的程序内容单独拿出来，当程序需要用到重复的内容时只要用“LCALL”指令后加上子程序的标号就可以了，可以参考（7）和（9）。（8）的意思是将 LED 变成高电平，发光二极管熄灭，和（6）是相反的，这样就可以达到闪烁效果了。（10）的含意是……还用我说吗，我的文章里也有子程序结构，哈哈。现在再回过头来以单片机的眼光看一遍从（5）到（10）的程序，是不是豁然开朗了，你兴奋的眼光开始闪烁了。是呀，（5）到（10）的程序形成了一个循环体，我们把它叫做线程，每一个单片机程序其实都是线程，只是现在我们写的是单线程，以后你还会接触到多线程，如果那时你还有兴趣继续学习的话。（11）和（12）之间的一大堆程序就是传说中的延时子程序了，延时的时间是 1 秒钟，如果你有空就将延时子程序中的“#8”改成“#4”并编译、下载到单片机上运行看看有什么变化。（13）也是伪指令，意思是说程序部分到此结束了，虽然你可以在 END 下面写一首抒情诗，但对于程序来讲没有任何意义。

#### [程序 2]

```
#include <AT89X52.h>           //你是 AT89S52 单片机           (1)
sbit LED = P1 ^ 0;             //你的 P1.0 引脚上接了一个发光二极管 (2)

void Delay(unsigned int a){     //延时程序(ms)                 (3)
    unsigned char i;
    while( --a != 0){
        for(i = 0; i < 125; i++);
    }
}

void main(void) {               //主函数（程序）                 (4)
    while (1) {                 //重复这些工作                 (5)
        LED = 0;                //点亮发光二极管             (6)
        Delay(1000);            //亮 1 秒钟（延时 1 秒钟）     (7)
        LED = 1;                //关掉发光二极管             (8)
        Delay(1000);            //关 1 秒钟                 (9)
    }
}
```

在介绍[程序 1]的时候我可费了不少心思，毕竟二年前我就已经和汇编说 88 了，有些地方还要回头翻笔记本才知道。下面要介绍[程序 2]了，全能 C 语言，一天写 8 遍，轻松加愉快。搞笑的是当我已经写了一半介绍文字时突然想起来，好像在《继往开来，后来居上》中的[程序 2]已经介绍过了，于是删除了这一段，这几天事情太多，稿子都写糊涂了，呵呵。看过了汇编的介绍，再加上你的聪明才智，你已经悟到程序的基本原理了吧，用这种眼光再看 C 语言程序就简单得多了。（4）是 main 函数，C 语言规定所有的程序都是从这里开始的，接口定义和子函数都是放在了 main 函数前面，其实放哪都可以，只是我习惯放在前面，因为前面不需要对子函数做声明，声明是什么呢，先假装不知道。现在只要知道汇编语言所谓的“程序”和 C 语言里讲的“函数”是一个意思就行了。

[程序 1]、[程序 2]，汇编语言、C 语言，当一切都展现在眼前时我们就可以不加个人情绪的对比、分析它们。下面我不说好坏，只是将特征介绍给你听。汇编语言可以设置程序的写入单片机的位置，C 语言不能；C 语言在程序后的文件大小要比汇编多，汇编编译后的文件大小容易掌控；[程序 2]的 C 语言程序只要小小改动就可以用在其它类型的单片机上面，[程序 1]的汇编程序只能用在 51 系列单片机上。汇编语言可以直接操作单片机全部硬件，C 语言不能完全操作。我喜欢汇编语言，我更爱 C 语言。

可能有一些朋友会说我讲的笼统，没讲伪指令和单片机指令集之类的，会不会误导初学者呀。是呀，想系统的学习单片机编程你可以随便在书店里单片机区的书架上抽一本书看到，论学理我的文章是打不过图书的，《无线电》一共才给了我这点版面，我只是用最通俗的方式带爱好者对编程产生兴趣并入门。至于有没有误导，我想读者的眼睛是闪闪发光的。

### 串口小秘书

当我们调试一个单片机程序时，总是希望可以看到程序运行的情况，这包括 I/O 口的状态和内部程序关键数值的变化情况。如果在黑黑的单片机芯片块上方开一个槽，装上一片 16 比 9 的真彩液晶屏，将单片机运行的状态和参数显示出来，那么作为使用者的我们一定非常愿意，可是买这样一块单片机非让你倾家当产不可。单片机的设计者们早已考虑到这个问题了，让编程人员可以轻松地与单片机交流的同时又不增加成本的最好方案就是串口。大家应该是熟悉电脑串口的，学名应该是叫做 RS232 接口，现在越来越多的家用电脑去掉了这个接口而改用性能更好的 USB 接口。好了，麻烦来了，我们需要买一个 USB 转 RS232 接口的转换器才可以拥有 RS232 串口，这是必须的工作，要知道这对于提高单片机程序的调试效率是多么重要。

以往的诸多单片机学习教材好像都不约而同的把单片机自带了串口功能当成了一个串口通信小实验来学习的，实验做完了，大家就忘记串口而将时间花在更有吸引力的液晶显示之类的实验中去，至少我见过的教材都是如此。目前爱好者们调试程序大多是采用功能设备的反应来诊断程序错误的，就像中医常用的望、闻、问、切四法。比如要让一个单片机的 I/O 口变成低电平，我们就会在这个 I/O 口上接入一个 LED 电路，在程序中设置这个 I/O 口的电平为低，当在程序运行后 LED 点亮了说明程序正确，否则就需要重新检查一下程序同时还要看电路是否无误。依我看来，这是最常见的方法但不是最好的，使用串口功能可以更爽的调试程序。同样将一个 I/O 口变成低电平只需要在程序中调试这个 I/O 口的电平为低，然后再读出这个 I/O 口的状态送到串口中，串口的另一端和电脑连接，I/O 口的状态将会显示在一个叫“超级终端”的软件上。就这样，有了串口就不需要实际的外围显示电路，而且再复杂的数据也可以轻松加愉快，实为单片机程序调试小秘书，工作学习必备佳品。“这么好的东西怎么用呢？”这位大哥问得好，下面我们就来瞅瞅这家伙咋用。

能使用这么好的串口工具统共分三步：设置好电脑终端软件、连接好硬件电路、在程序中加入串口程序。

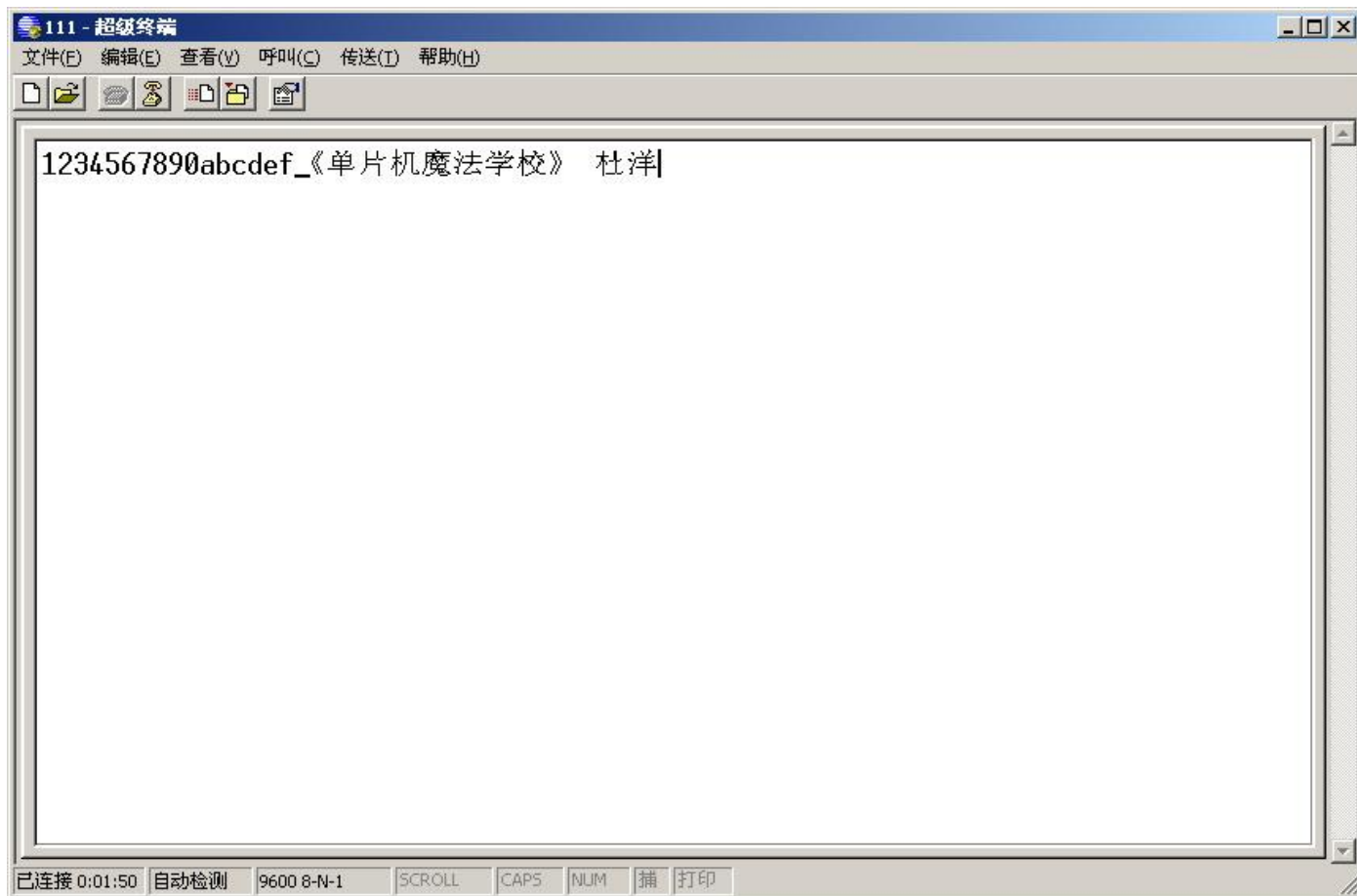


图 1

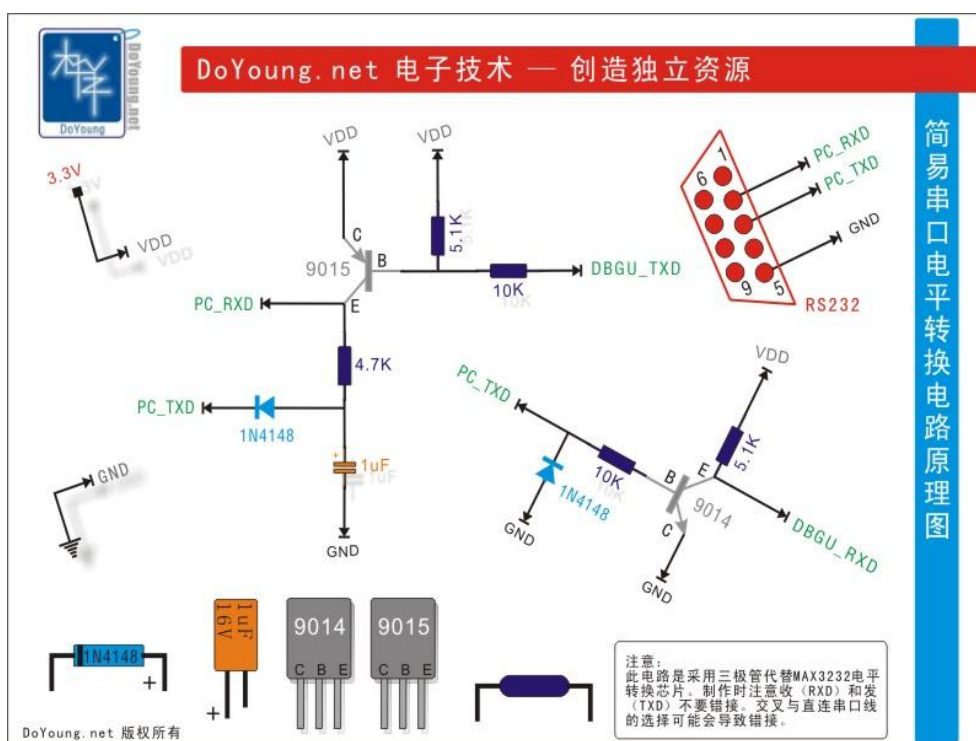


图 2



第一步，设置电脑终端软件。第一次听到“超级终端”这个词是我刚参加工作不久，当时是为了看一块开发板系统的启动信息，同事怔怔的看着我，我却以为这是一款独立的软件工具，还正要到网上搜索。同事过来点击桌面左下角的“开始→所有程序→附件→通讯→超级终端”的时候我算是彻底傻眼了，学了这么久单片机还在四处下载串口助手之类的软件，没想到“天生”的串口助手就在我的眼皮底下。超级终端是一款终端间的通信软件，当然是包括电脑与电脑的通信也包括电脑和其它设备的通信。这是我们天然的软件工具，除非你有更高的追求，否则超级终端是可以满足一般实验的。首次启动超级终端会出现一些关于调制解调器的拨号设置，这是为拨号网络通信时而设计的，这里我们点击取消即可。随后在弹出“新建连接”的窗口中我们需要给超级终端取一个名字并选择一个个性图标。在弹出的“连接到”窗口的“连接时使用(N):”下拉框中选择你连接到单片机上对应的串口号，我这里选择的是“COM2”，点击“确定”后会弹出“COM2 属性”窗口，在“端口设置”选项卡中设置“每秒位数”是4800（波特率），“数据位”是8，“奇偶校验”是无，“停止位”是1，“数据流控制”是无，然后点击“确定”。这时我们已经完成了电脑软件的准备工作，刚才的设置项目可以在超级终端菜单栏的“文件→属性”中修改，在修改之前一定要点击菜单栏中的“呼叫→断开”，设置完成后再点击“呼叫→呼叫”，否则将不能设置或正常通信。

设置好了，怎么来测试一下呢，当然在没有完成后面的制作时也是可以测试的，正是为了可以让大家每一步都可以验收自己的阶段性成果，我在制作顺序上精心安排。现在用一根小导线将电脑串口端的第2和第3针短接，目的是为了电脑串口发送的数据返回给自己，这样我们就可以在超级终端窗口中输入字符而在窗口中看到我们输入的字符了，这相当于一个写字板软件的功能，如果可以看到我们输入的字符则证明我们的设置是成功的，如图1所示。不然就需要检查串口号是否正确，导线短接是不是第2和第3针，串口针的编号在接口上可以看到的。测试一旦成功我们就可以往下继续了，不过我建议你有空时像玩KEIL软件一样玩一玩超级终端，要知道今后的嵌入式系统学习中它饰演着重要角色。

第二步，连接硬件电路。电脑上没有9针串口的朋友可以到电脑配件市场上买到一种USB转RS232的转接器，我现在用的笔记本就是用的这种方式，还算方便，把USB转RS232的转接器和串口电平转换电路放在一个小盒子里，乍一看去还以为我是用USB接口与单片机通信呢，其实没那么牛叉。如果你的电脑上正好有一个串口接头就是最好了，不过别以为找几条导线把串口中的线拉出来接到单片机的接口上就行了，如果真这么简单我也不需要占用一节文章介绍串口了，不过说实话我还真不知道直接从串口拉线到单片机的接口会有什么现象，有兴趣的兄弟可以试一试，然后写信告诉我。通常我们都是需要一种电平转换电路的，单片机电路是TTL电平，串口是RS232电平。我们有一款电路可以实现电平转换，最常用的一种是专用的电平转换芯片（如MAX232）实现的转换电路，这种电路稳定性高但成本也高，一片MAX232芯片市场价在4元左右。还有一种光藕电平转换，这是我在国外的制作网站上看到的电路，它巧妙的运用了光藕元器件实现了电气隔离的电平转换，其电路所用元器件很少，也有保护电路的作用，只是光藕市场价也不便宜。最便宜的当属分立元件制作的电路了，只有电阻、二极管、三极管，购买容易、制作方便，不过性能不如前两个电路，对于我们一般的实验已经足够了。

采用MAX232芯片制作电平转换电路的例子在《无线电》和其它文章中太多了，我要给大家来点个性的。图2所示是采用分立元件制作的串口电平转换电路，我已经制作出来了，效果很理想。在制作的过程中，RXD（接收）和TXD（发送）端不要接反，电脑上的RXD要接到单片机上的TXD才对，余下的就看你的电路制作水平了，电路这么简单，出了问题自己解决吧，不过这里还是要讲一下阶段验收方法。和前面一样，还是用刚才那根导线在单片机的一端将RXD和TXD短接，给实验板和转换板电路接通电源，再在超级终端窗口中输入字符，成功时是会有返回显示的，如果没有回显的话除了检查转换电路还要看看电脑端的TXD和RXD有没有接反。知道吗，我在写文章的时候总是会考虑许多制作时可能遇见的问题，我尽量多说一些，不过最重要的还是要你自身具有逻辑推理能力和经验，如果你没有它们那现在正是锻炼的好机会，最怕的就是不加思考的问人，要知道从长远角度看这对任何人都没有好处，在网络技术如此发达的今天，我认为拿一个没有建设性的问题四处寻问是不明智的。

第三步，在程序中加入串口程序。这是最后的环节了，把一个串口程序加入到我们要调试的程序之中，当程序运行时我们所需要的数据就在串口得到了，但现在我们先来测试一下硬件连接吧。保持前两步的状态，并在本期《无线电》光盘里找到 HEX 文件夹下的“串口测试程序.hex”文件，下载到单片机里，单片机电路是《单片机入门制作专辑》中的实验板电路，有一个 LED 和一个按键，晶振是 12MHz。我们再在超级终端里输入字符就会有对应的字符显示在窗口中，这好像与第二步的测试相同，如果 RXD 和 TXD 短路也会出现这种效果的。所以你还要多实验一步，就是将超级终端的每秒位数（波特率）设置到 9600 之后再试一下，如果输入字符没有显示，则说明硬件没有短路，串口测试成功，将波特率再调回到 4800，我们取得了又一次阶段性成功。[程序 3]是串口测试程序的内容，看懂或看不懂都是正常的，我们现在的目的是应用，深入研究就留给你茶余饭后慢慢把玩吧。

### [程序 3]

```
#include <reg52.h> //包函 8051 内部资源的定义（与 AT89X52.h 功能基本相同）
```

```
void Init_COM(void){
    TMOD = 0x20;    //定时器工作方式设置，定时器 1 方式 2
    SCON = 0x50;    //串口工作方式 1，允许串口接收
    TH1 = 0xF3;     //定时器初值高 8 位设置
    TL1 = 0xF3;     //定时器初值低 8 位设置
    PCON = 0x80;    //波特率倍频
    TR1 = 1;        //定时器启动
}

void main(){
    unsigned char dat;
    Init_COM();      //串口初始化
    while(1){
        if (RI == 1){    //接收中断标志位为 1 时
            dat = SBUF;   //接收数据 SBUF 为单片机的接收发送缓冲寄存器
            RI = 0;       //令接收中断标志位为 0（软件清零）
            SBUF = dat;   //将接收的数据发送回 PC
        }
    }
}
```

### [程序 4]

```
#include <AT89X52.h> //你是 AT89S52 单片机
sbit LED = P1 ^ 0;   //你的 P1.0 引脚上接了一个发光二极管

void Init_COM(void){    //串口初始化程序，波特率 4800 (1)
    TMOD = 0x20;        //定时器工作方式设置，定时器 1 方式 2
    SCON = 0x50;        //串口工作方式 1，允许串口接收
    TH1 = 0xF3;         //定时器初值高 8 位设置
    TL1 = 0xF3;         //定时器初值低 8 位设置
    PCON = 0x80;        //波特率倍频
```

```
    TR1 = 1;                //定时器启动
}

void Delay(unsigned int a){    //延时程序(ms)
    unsigned char i;
    while( --a != 0){
        for(i = 0; i < 125; i++);
    }
}

void main(void) {            //主函数（程序）
    unsigned char c=0;        //设置一个变量 C，初值为 0，假设其为重要数据        (2)
    Init_COM();               //串口初始化                                (3)
    while (1) {               //重复这些工作
        LED = 0;              //点亮发光二极管
        Delay(1000);          //亮 1 秒钟（延时 1 秒钟）
        LED = 1;              //关掉发光二极管
        Delay(1000);          //关 1 秒钟
        SBUF = c + 0x30;       //将调试数据发送回 PC，0x30 是数字显示的偏移量        (4)
        c++;                  //变量 c 的值加 1                                (5)
        if(c > 9){             //如果 c 的值大于 9 则回到 0 重新计数                (6)
            c = 0;             //让 c 等于 0
        }
    }
}
```

[程序 4]是加入了串口小秘书的闪烁 LED 灯的程序，因为这个闪烁 LED 的程序实在是没有什么有价值的数 据，不过我又不好举其它例子，这样会显得太突然，最后我就设计了一个数据出来，用来计算程序循环运行的次数，当程序循环运行一次则变量 c 加 1，当 c 加到 9 时再回到 0 继续加 1。最终程序的数据在串口上显示出来，这是我们的目的所在，日后会有复杂的程序调试，借此方法是可以抛砖引玉的。看看我们在原来的闪烁 LED 基础上加入了什么内容才会让数据在串口中显示，(1)是我们见过的串口初始化程序，这是必不可少的。(3)是在主程序中调用初始化程序，在主程序一开始加入这段后串口就开始工作了。(4)是发送数据到串口上，将我们需要的数据送到 SBUF 里面就相当于送给了串口发送到 PC 机了，也是我们实现目的的关键一步。(2)、(5)、(6)都是我们无中生有的重要数据相关的部分，就不多费口舌。值得注意的是超级终端窗口所显示的是 ASCII 码，而我们变量 c 的值是十进制数据，如果直接送去串口只会显示一些奇怪字符，如果你是 ASCII 码高手当然能看出意思来，幸好我不是所以加了 0x30 的一个偏移量，这可以让 c 的值正好对应 ASCII 码中的数字区。我还写了一篇《有趣的超级终端玩法》的文章，虽然没有在杂志上发表但在网上是可以找到的，我在文章中介绍了超级终端更高级的功能，还有一些参考程序，如果你有兴趣不妨搜索去吧。当然串口的收发功能还可以在电脑的串口上还可以输入数据给单片机，[程序 3]可以给你启示，总之在串口小秘书这里我介绍的是方法，很有用吧？至少当你了解之后再学习 ARM9 中的调试串口概念时就不会陌生。上面所有涉及的程序内容在本期《无线电》光盘里都可以找到，而且是我精心制作，全心全意为读者服务。



## 改革创新

创新和创意是外来词，中国古代没人会说：你这个东东好有创意哦！现代社会人们被创新这条疯狗追得喘不上气来，而真正优秀的创新却没有几个。大多数人认为的创新是指前所未有、与众不同的，我认为这应该叫狭义的创新，而我认为的广义创新除了狭义创新之外还包括借鉴的创新和优化的创新。也就是说使用别人的程序为自己所用是创新，把现有的程序变的更好也是创新。就好像我的这篇文章的写作风格在电子技术领域算是与众不同了，大家都看出这是创新了，你看了我的文章之后把程序写的更细致更稳定却不一定被认为是创新。这只是因为前者更容易被注意，而后者不被大家认同。

好，如果你想制作一台电子钟，看看你有哪些方法。最简单的一种，买一个现成的电子钟在正面贴上自己的名字，然后说是自己制作的，如果你不认为这很可耻的话，应该是最简单的方法。再有就是找到别人制作电子钟的资料，依原理仿制一个，再修改一下电路和程序让它更具个性。还有就是不参考任何资料，凭着自己高超的创造力完成制作。最前一种方法没什么可讲，只要说谎时脸色不红不白就够了。最后一种方法要求不参考任何资料，当然也包括本文了，所以我讲什么都没有意义。剩下借鉴前人的资料完成制作，这个我比较在行，这里不吐不快。

再回头看看[程序 2]，要知道我们有多少发挥的空间不一定靠了解每一条程序的用意，只要抓住几处关键点就可以任由想象。这是一个 LED 闪烁的程序，我们的局限就在 LED 的闪烁方式的创新，因为你不可能把它修改成一台电子琴的程序，不仅硬件不允许，而且软件会和现在的完全不同，如果想制作电子琴就要先找到与电子琴类似的硬件和程序，不然就和相声《做大褂》一样，一丈三的蓝布本想做褂却补了袜子。(2) 是 LED 在单片机上的接口定义，P1^0 表示程序将控制 P1.0 接口上的 LED，如果我们想控制其它接口上的 LED 可以改成 P1^1 或者其它对应的接口代号。(6) 到 (9) 是程序运行的主要部分，单片机的循环任务就在 while(1){ } 的大括号中间，我们创新的爆破点就在于此。Delay( ) 括号里的数值可以在 0 到 65535 之间改动，表示了延时的时间。LED = 0 和 LED = 1 是将控制接口输出低电平和高电平的过程。了解这些之后你想到了什么，有什么新想法、新创意？要知道这年头搞技术的拼得就是精致和创意，一个 LED 你能玩出花来也算你是高手了，因为在没有任何参考的前提下我设计出了渐变亮度的闪烁 LED 灯程序，填补了我两年前自以为是高手却拿不出原创作品的空白。我相信你比我更有创意，但要拿作品说话，非常希望和诸位多交流。

[程序 5] 就是我号称渐明渐暗闪烁 LED 灯的程序，有人以为我用了专用的硬件电路，还有说我是采用的模拟信号，虽然是诽谤但我也很高兴。其实原理并不难，利用了占空比的概念，在[程序 2] 的闪烁 LED 的基础上让延时的时间可以逐渐变化，因为变化很快所以人眼的感觉就成了亮度的变化。现在这个程序在爱好者中已经不是什么稀奇，不过思路依然需要了解。我不做更多解释，现在是考核你分析能力的时候，想想你独到的创意，再用心灵手巧实现它，玩单片机编程是多么快乐的事情呀，学习过程是很让人兴奋的，我经历过你现在的心情所以我由衷为你高兴，祝你好运我的朋友！

## [程序 5]

```
#include <AT89X52.h>
sbit LED = P1 ^ 0;           //你的 P1.0 引脚上接了一个发光二极管

void delay (unsigned int d){   // 延时功能函数--由 d 决定延时长度
    unsigned char i;
    while( --d != 0){
        for(i = 0; i < 2; i++);
    }
}
```

```
void main (void){           // 彩灯主函数，实现彩灯的显示
    unsigned char a, b;     // 使用两个变量，a 控制延时长度，b 控制循环周期
    while (1){
        LED = 1;
        a = 0x00;
        for (b=250; b>0; b--){    // 定义半周期循环时间（次数）
            a++;
            delay (a);             // 延时长度随 a 而改变
            LED = 0;
            a = ~a;                // a 值取反，决定灯灭时的占空比
            delay (a);
            LED = 1;
            a = ~a;                // a 值取反，使 a 回到原值继续循环
        }
        LED = 0;                // 灯亮，使灯达到最亮
        a = 0x00;
        for (b=250; b>0; b--){
            a++;
            delay (a);
            LED = 0xff;
            a = ~a;
            delay (a);
            LED = 0;
            a = ~a;
        }
        LED = 1;
    }
}
```

#### 版本信息

题目 单片机编程魔法学校

作者 杜 洋

时间 2008.3.1

版本 V1.0

声明 本站内容（包括程序代码、文档、照片、视频等）属个人所有，未经网站作者同意请勿转载或引用，对于转载或复制而造成的任何不良后果概不负责。对于本站内免费下载的资料、图片及视频不能保证其真实可靠，对于免费下载的程序代码本站作者不给予技术支持和服务。

本文版权属《无线电》杂志所有，DoYoung.net 经特许转载！