

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

前言	第 1 页
LED 数码管实验原理图	第 4 页
加、减计数器源程序	第 5 页
24 小时时钟源程序	第 7 页
倒记时定时器源程序	第 11 页
串行接口 LED 数码管显示器源程序	第 15 页

前言

对初学者来说，单片机是一个很难把握的东西。硬件与软件的理论基础需要耗费很多的时间才能掌握，而另一方面，即使已经学习了大量的软、硬件基础理论，在进行具体的系统设计时仍然会觉得无从下手。学习单片机除了掌握理论知识外，必须进行实战。在积累大量理论与实战的经验后才能得心应手的使用单片机。然而，对业余爱好者来说，单片机的开发系统仍然十分昂贵。制做一块理想的单片机实验板也非易事。

毫无疑问，MCS-51 系列单片机仍然是最理想的学习用单片机，其内部资源丰富，样片容易得到，学习资料也非常丰富。C 语言编译器及片内 Flash 程序存储器的使用降低了进行单片机实验的门槛。以下的一些实验全部基于 ATMEL 的 AT89C2051。AT89C2051 相当于 80C31+74HCT373+27C16，20 脚双列直插（DIP）封装，仅需连接晶振及电源即可工作。本实验的基本系统为 AT89C2051 通过动态扫描驱动 6 位 LED 数码管及 6 个按键。整个电路（见下图）十分简洁，适合业余爱好者自行搭制。以下几个趣味实验均使用该硬件完成，通过这些实验可掌握一些单片机系统设计的常用方法，以及 MCS-51 内部资源的使用方法。



一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

(1) 加、减计数器

具有清零，+1/-1 按钮，可接开关或继电器触点等，具有去抖动功能。记数范围 0-999999，无用的零自动消隐。

(2) 24 小时时钟

可设置时间，压调整按钮进入调整状态，依此选择要调整的小时、分钟或秒，再次按压调整按钮退出调整状态。进入调整状态后，所要调整的项目闪烁，同时，时钟继续运行，并不会被中断。可通过依此按压调整按钮退出调整状态，也可在停止调整 10 秒后自动退出调整状态。

(3) 倒记时定时器

可设置倒记时的时间，单位为秒，范围 000000-999999 秒，约 11.5 天。压调整按钮进入调整状态，依此调整第 1-6 位，再次按压调整按钮退出调整状态，也可在停止调整 10 秒后自动退出调整状态。在进入调整状态后，相应要被调整的位闪烁。倒计时至 0 时由 P10 输出 1s 的负脉冲。

(4) 串行接口的 6 位 LED 数码管显示器

通过串行口设置要显示的内容，可作为独立的显示模块，稍做修改可实现 4 位显示同时支持 4 个按钮的功能。加上 MAX232 可直接与 PC 的 RS-232C 串行口连接。在这一个实验中，程序实现一个 24 小时时钟，其时间可通过串行口设置，串行口设置为 9600, 8, N, 1。设置方法如下：

#hhmmss<cr> 即以#作为起始位，紧随其后的为小时、分钟和秒钟的十进制数对应的 ASCII 码，并以回车键结束。例如：#102638<cr> 设定当前时间为 10 点 26 分 38 秒。

在这一个实验中，使用了定时器中断和串行口接收中断，关于 MCS-51 和 PC 机串行收发的进一步资料可参考 Boan 单片机设计室(<http://boanmcu.yeah.net/>)的开发工具中的有关章节。

在这些实验中，使用 P10-P17 驱动数码管的 8 个段，P31-P37 分别用来选择 6 个数码管，数码管的电源可直接与单片供电接在一起，即将 J1 的第 2 脚与第 3 脚短接，P30 既可作为串行口使用，也可与 P31-P37 构成 6 个扫描按钮。在实验 1、2 和 3 中，分别使用 SW6、SW3 和 SW1 分别作为调整（清零）按钮及+1/-1 按钮，在实验 4 中 P30 作为串行口使用。数码管的动态刷新与键盘的动态扫描均在定时器中断中完成，定时器每 2.5ms 中断一次，6 位数码管刷新一次需要 $2.5 \times 6 = 15\text{ms}$ ，刷新频率为 66.7Hz，因此

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

显示效果非常稳定。同时，键盘也按 15ms 的速度扫描，这就有效的去除了抖动。其实，使用定时器中断去除抖动较软件循环延时更为有效，软件循环延时既耗费 CPU 时间，同时也严重影响 CPU 对其它事件的响应。顺便说一句，定时器中断对于软件设计极其重要，除了以上所讲的动态扫描外与键盘去抖动外，使用定时器中断可同时实现任意多个软件延时，而且，对于多任务及软件的可使用性都非常有帮助。

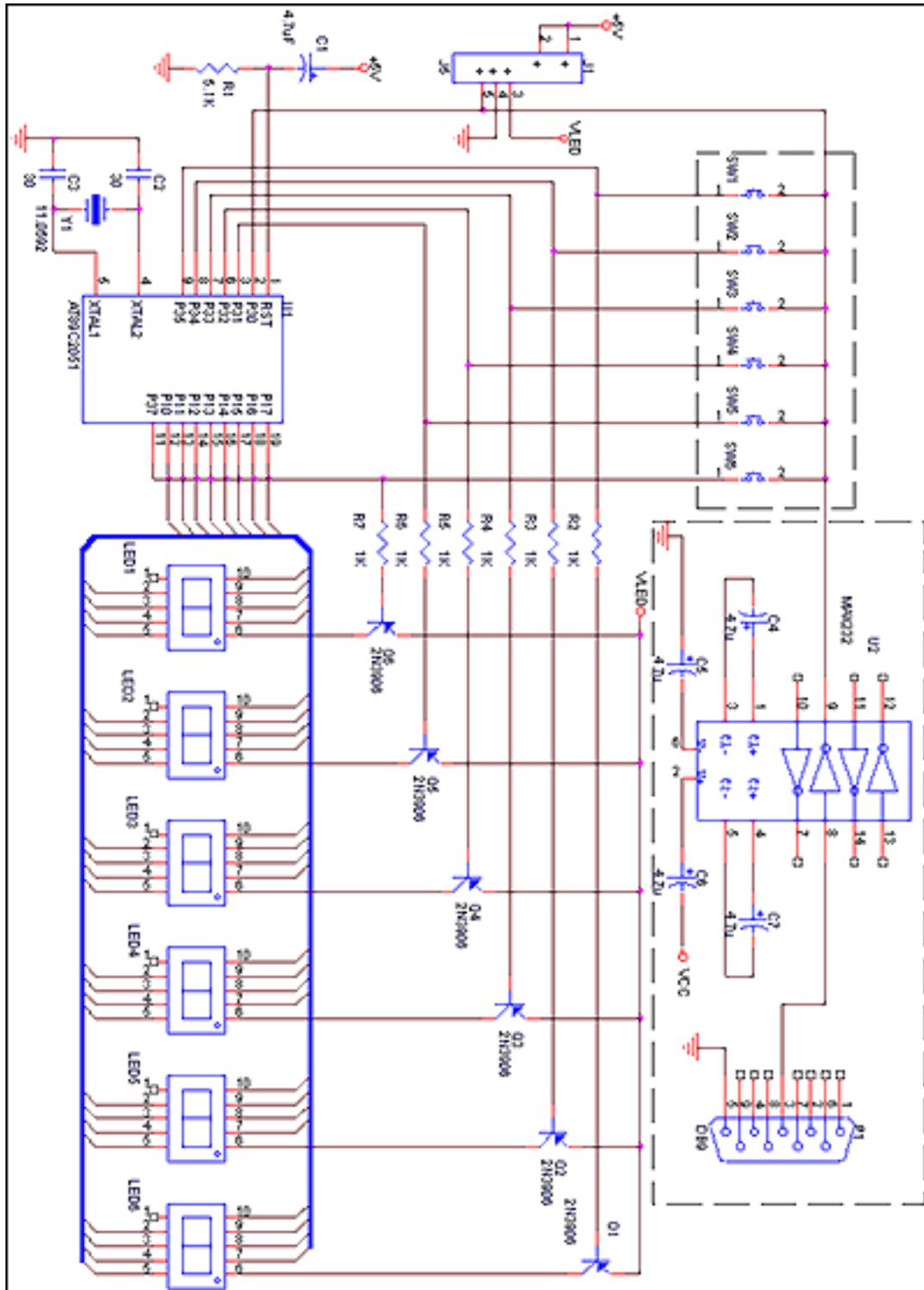
在常见的一些单片机教材中，也有多方面的实验，然而，大多数仅给出原理性的分析和程序片段，而且这些实验侧重于某一点的应用而忽视了系统软件的整体观念。同时也极其乏味。这不利于学习与提高。而以上这些试验均强调其趣味性，并附有完整的源程序及编译后的可执行代码，这些程序中使用的均为典型的单片机系统设计方法，使用这些方法编写的程序可获得很好的性能，同时，尽可能使用浅显易懂的方法表述。这些是这组趣味实验的主要特点。通过这些实验可以学习定时/计数器，串行口及中断的使用，键盘扫描程序的编制及键盘去抖动的方法。以及数码管的动态扫描显示。通过这些程序也可以学习到如何合理分配单片机的资源，以及基于消息驱动的软件设计思想，哪些任务适合由中断服务程序处理，哪些任务适合由主程序处理。消息的产生与处理，以及定时器的使用与多任务的实现。这些程序均由 Franklin C-51 V5.02 for DOS 语言写成，生成代码很短。

以上的这些实验的源程序和二进制代码可到 Boan 单片机设计室下载，也可在 21IC 的用户自由发布区下载其 PDF 文件，Boan 单片机设计室将在近期提供更多的一些适合初学者的一些趣味制作，敬请留意。

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

LED 实验板原理图如下：



一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

1. 加、减计数器源程序: counter.c

```
#include "stdio.h"
#include "reg51.h"

// 0 - 9 的编码
unsigned char code table[] = { 0x11, 0xb7, 0x29, 0x23, 0x87, 0x43, 0x41, 0x37, 0x01, 0x03, 0x00 };

//第 1—6 位的片选线
unsigned char code dig[] = { 0x7f, 0xfd, 0xdf, 0xf7, 0xef, 0xfb };
unsigned char buffer[8];
unsigned char scan, i, ch, KeyVal;
bit          fKey, Key1, Key2, Key3, OldKey1, OldKey2, OldKey3, fBI;
unsigned long int cn, tmp;

void  init(void);

main()
{
  init();

  while(1) {
    if(fKey) {           // 有按键按下?
      fKey = 0;         // 清除按件标志, 并处理该按键

      if(KeyVal==1) cn = 0;           // 清零按钮, 计数值清零
      if(KeyVal==2) cn++;            // +1 按钮, 计数值+1
      if(KeyVal==3 && cn!=0) cn--;    // -1 按钮, 计数值-1

      EA = 0;                    // 计算显示结果
      tmp = cn;
      for(i=0; i<6; i++) {
        buffer[5-i] =(unsigned char) ( tmp % 10 );
        tmp = tmp / 10;
      }
      EA = 1;
    }
  }
}
```

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

```
void init(void) {

    scan = 0; KeyVal = 0; cn = 0;
    fKey = 0; fBI = 1;
    Key1 = Key2 = Key3 = 0;
    OldKey1 = OldKey2 = OldKey3 = 1;

    for(i=0;i<6;i++) buffer[i] = 0;

    TCON &= 0xCF;           // 初始化 Timer0
    TMOD &= 0xF0;
    TMOD |= 0x01;
    TH0 = TL0 = 0;
    EA   = 1;
    ET0  = 1;
    TR0  = 1;

}

timerint () interrupt 1 {

    TR0 = 0;
    TL0 = 0x11;
    TH0 = 0xf7;           // 0xf70c for 2.5ms
    TR0 = 1;

    if(scan==1 && !fKey) { // 扫描清零按钮
        Key1 = P30;
        if(!Key1 && OldKey1) { fKey = 1; KeyVal = 1; }
    }

    if(scan==3 && !fKey) { // 扫描+1 按钮
        Key2 = P30;
        if(!Key2 && OldKey2) { fKey = 1; KeyVal = 2; }
    }

    if(scan==4 && !fKey) { // 扫描-1 按钮
```

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

```
Key3 = P30;
if(!Key3 && OldKey3) { fKey = 1; KeyVal = 3; }
}

OldKey1 = Key1; OldKey2 = Key2; OldKey3 = Key3;

P1 = 0xff; P3 = dig[scan];
ch = buffer[scan];           // 依次显示 6 位，并消隐无效 "0 "
if(ch==0) {
    if(fBI && scan!=5) P1 = 0xff; else P1 = table[ch];
} else { fBI = 0; P1 = table[ch]; }

scan++; if(scan==6) { scan = 0; fBI = 1; }
}
```

2. 24 小时时钟源程序: clock.c

```
#include "stdio.h"
#include "reg51.h"

// 0 - 9 的编码
unsigned char code table[] = { 0x11, 0xb7, 0x29, 0x23, 0x87, 0x43, 0x41, 0x37, 0x01, 0x03, 0x00 };
unsigned int count2;

// 第 1—6 位的片选线
unsigned char code dig[] = { 0x7f, 0xfd, 0xdf, 0xf7, 0xef, 0xfb };
unsigned char buffer[8];
unsigned char count1, scan, i, ch, hour, min, sec, KeyVal, modify, OverTime;
bit fKey, fSec, Key1, Key2, Key3, OldKey1, OldKey2, OldKey3, blink;

void init(void);
void display(void);

main()
{
    init();
```

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

```
while(1) {

    if(fSec) {                // 秒标志被置位?
        fSec = 0;            // 清除秒标志

        sec++;                // 计算当前时间
        if(sec==60) {
            sec = 0; min++;
            if(min==60) {
                min = 0; hour++;
                if(hour==24) hour = 0;
            }
        }

        display();           // 显示当前时间
    }

    if(fKey) {                // 有按键按下?
        fKey = 0;            // 清除按件标志，并处理该按键

        if(KeyVal==1) {      // 是调整按钮?
            modify++;         // 进入相应状态，并重置超时定时器
            if(modify==4) { modify = 0; OverTime = 0; } else OverTime = 10;
        }

        if(KeyVal==2 && modify!=0) { // +1 按钮，小时、分钟或秒+1
            if(modify==1) { hour++; if(hour==24) hour = 0; }
            if(modify==2) { min++; if(min==60) min = 0; }
            if(modify==3) { sec++; if(sec==60) sec = 0; }
            display(); OverTime = 10; // 刷新显示，并重置超时定时器
        }

        if(KeyVal==3 && modify!=0) { // -1 按钮，小时、分钟或秒+1
            if(modify==1) { if(hour!=0) hour--; }
            if(modify==2) { if(min!=0) min--; }
            if(modify==3) { if(sec!=0) sec--; }
        }
    }
}
```

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

```
    display(); OverTime = 10;          // 刷新显示，并重置超时定时器
}

}

}
}
```

```
void init(void) {
```

```
    count1 = count2 = 0; scan = 0; KeyVal = 0;
    hour = min = sec = 0;
    fKey = 0; modify = 0; blink = 0; OverTime = 0;
    Key1 = Key2 = Key3 = 0;
    OldKey1 = OldKey2 = OldKey3 = 1;
```

```
    for(i=0;i<6;i++) buffer[i] = 0;
```

```
    TCON &= 0xCF;                      // 初始化 Timer0
    TMOD &= 0xF0;
    TMOD |= 0x01;
    TH0 = TL0 = 0;
    EA   = 1;
    ET0  = 1;
    TR0  = 1;
```

```
}
```

```
timerint () interrupt 1 {
```

```
    TR0 = 0;
    TLO = 0x12;
    TH0 = 0xf7;                          // 0xf712 for 2.5ms
    TR0 = 1;
```

```
    if(scan==1 && !fKey) { // 扫描调整按钮
        Key1 = P30;
        if(!Key1 && OldKey1) { fKey = 1; KeyVal = 1; }
```

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

```
}

if(scan==3 && !fKey) {    // 扫描+1 按钮
    Key2 = P30;
    if(!Key2 && OldKey2) { fKey = 1; KeyVal = 2; }
}

if(scan==4 && !fKey) {    // 扫描-1 按钮
    Key3 = P30;
    if(!Key3 && OldKey3) { fKey = 1; KeyVal = 3; }
}

OldKey1 = Key1; OldKey2 = Key2; OldKey3 = Key3;

P1 = 0xff; P3 = dig[scan];
ch = buffer[scan];        // 依次显示 6 位

if(modify==0) P1 = table[ch];
else if(modify==1 && blink && (scan==0 || scan==1) ) P1 = 0xff;
else if(modify==2 && blink && (scan==2 || scan==3) ) P1 = 0xff;
else if(modify==3 && blink && (scan==4 || scan==5) ) P1 = 0xff;
else P1 = table[ch];

scan++; if(scan==6) scan = 0;

count1++;
if(count1==200) { count1 = 0; blink = ~blink; }    // 0.5s 闪烁定时器

count2++;
if(count2==400) {
    count2 = 0; fSec = 1;                            // 秒脉冲标志置位

    if(OverTime) { OverTime--; if(OverTime==0) modify = 0; }    // 10s 超时定时器
}

}

void    display(void) {
```

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

```
buffer[0] = hour/10; buffer[1] = hour%10;    //    显示 hh mm ss
buffer[2] = min/10;  buffer[3] = min%10;
buffer[4] = sec/10;  buffer[5] = sec%10;
}
```

3. 倒记时定时器源程序: timer.c

```
#include "stdio.h"
#include "reg51.h"

// 0 - 9 的编码
unsigned char code table[] = { 0x11, 0xb7, 0x29, 0x23, 0x87, 0x43, 0x41, 0x37, 0x01, 0x03, 0x00 };
unsigned int  count2;

//第 1—6 位的片选线
unsigned char code dig[]  = { 0x7f, 0xfd, 0xdf, 0xf7, 0xef, 0xfb };
unsigned char buffer[8], val[6];
unsigned char count1, scan, i, ch, KeyVal, modify, OverTime;
bit          fKey, fSec, Key1, Key3, OldKey1, OldKey2, OldKey3, blink, fTimer, fSTOP;
unsigned long int cn, tmp;

void  init(void);
void  display(void);
void  Cal_cn(void);
void  Cal_val(void);

main()
{
    init();

    while(1) {

        if(fSec) { // 秒标志被置位?
            fSec = 0;                // 清除秒标志
            if(!fSTOP && modify==0) {
                if(cn!=0) cn--; else { fTimer = 1; fSTOP = 1; }
            }
        }
    }
}
```

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

```
    display();
}

if(fKey) {                // 有按键按下?
    fKey = 0;             // 清除按件标志, 并处理该按键

    if(KeyVal==1) {      // 是调整按钮?
        fSTOP = 0; modify++; // 进入相应状态, 并重置超时定时器
        if(modify==7) { modify = 0; OverTime = 0; } else OverTime = 10;
    }

    if(KeyVal==2 && modify!=0) { // +1 按钮, 相应位+1
        Cal_val();
        if(val[modify-1]<9) val[modify-1]++;
        Cal_cn();
        display(); OverTime = 10; // 刷新显示, 并重置超时定时器
    }

    if(KeyVal==3 && modify!=0) { // -1 按钮, 相应位-1
        Cal_val();
        if(val[modify-1]>0) val[modify-1]--;
        Cal_cn();
        display(); OverTime = 10; // 刷新显示, 并重置超时定时器
    }

}

}

}

void init(void) {

    count1 = count2 = 0; scan = 0; KeyVal = 0;
    cn = 0; for(i=0;i<6;i++) val[i] = 0;
    fKey = 0; modify = 0; blink = 0; OverTime = 0; fTimer = 0; fSTOP = 0;
    Key1 = Key2 = Key3 = 0;
    OldKey1 = OldKey2 = OldKey3 = 1;

    for(i=0;i<6;i++) buffer[i] = 0;
```

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

```
TCON &= 0xCF;           // 初始化 Timer0
TMOD &= 0xF0;
TMOD |= 0x01;
TH0 = TL0 = 0;
EA   = 1;
ET0  = 1;
TR0  = 1;

}

timerint () interrupt 1 {

    TR0 = 0;
    TL0 = 0x12;
    TH0 = 0xf7;           // 0xf712 for 2.5ms
    TR0 = 1;

    if(scan==1 && !fKey) { // 扫描调整按钮
        Key1 = P30;
        if(!Key1 && OldKey1) { fKey = 1; KeyVal = 1; }
    }

    if(scan==3 && !fKey) { // 扫描+1 按钮
        Key2 = P30;
        if(!Key2 && OldKey2) { fKey = 1; KeyVal = 2; }
    }

    if(scan==4 && !fKey) { // 扫描-1 按钮
        Key3 = P30;
        if(!Key3 && OldKey3) { fKey = 1; KeyVal = 3; }
    }

    OldKey1 = Key1; OldKey2 = Key2; OldKey3 = Key3;

    P1 = 0xff; P3 = dig[scan];
    ch = buffer[scan];           // 依次显示 6 位
```

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

```
if(modify==0) {
    if(fTimer) P1 = table[ch] & 0xfe; // 倒计时结束，启动 P10 (P10 低电平)
    else P1 = table[ch];
}
else if(modify==1 && blink && scan==0) P1 = 0xff;
else if(modify==2 && blink && scan==1) P1 = 0xff;
else if(modify==3 && blink && scan==2) P1 = 0xff;
else if(modify==4 && blink && scan==3) P1 = 0xff;
else if(modify==5 && blink && scan==4) P1 = 0xff;
else if(modify==6 && blink && scan==5) P1 = 0xff;
else P1 = table[ch];

scan++; if(scan==6) scan = 0;

count1++;
if(count1==200) { count1 = 0; blink = ~blink; } // 0.5s 闪烁定时器

count2++;
if(count2==400) {
    count2 = 0; fSec = 1; fTimer = 0; // 秒脉冲标志置位，并关闭 P10 输出

    if(OverTime) { OverTime--; if(OverTime==0) modify = 0; } // 10s 超时定时器
}
}

void Cal_cn(void) {

    cn = 0;
    for(i=0;i<6;i++) {
        cn *= 10; cn += val[i];
    }
}

void Cal_val(void) {

    tmp = cn;
    for(i=0;i<6;i++) {
```

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

```
    val[5-i] =(unsigned char) ( tmp % 10 );
    tmp = tmp / 10;
}

}

void    display(void) {

    tmp = cn;
    for(i=0;i<6;i++) {
        buffer[5-i] =(unsigned char) ( tmp % 10 );
        tmp = tmp / 10;
    }

}
```

4. 串行接口 LED 数码管显示器源程序: rs232.c

```
#include "stdio.h"
#include "reg51.h"

#define MAXLEN  8           // 串行口缓冲区 8ytes 大小

// 0 - 9 的编码
unsigned char code table[] = { 0x11,0xb7,0x29,0x23,0x87,0x43,0x41,0x37,0x01,0x03,0x00 };

//第 1—6 位的片选线
unsigned char code dig[]   = { 0x7f,0xfd,0xdf,0xf7,0xef,0xfb };
unsigned char buffer[8],cbuf[8];
unsigned char count1,scan,i,ch,hour,min,sec,ptr;
bit          fSec;
unsigned int  count2;

void    init(void);
void    display(void);

main()
{

    init();

    while(1) {
```

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

```
if(fSec) {           // 秒标志被置位?
    fSec = 0;         // 清除秒标志

    sec++;           // 计算当前时间
    if(sec==60) {
        sec = 0; min++;
        if(min==60) {
            min = 0; hour++;
            if(hour==24) hour = 0;
        }
    }

    display();       // 显示当前时间
}

}

}

void init(void) {

    scan = 0; fSec = 0; ptr = 0;

    for(i=0;i<6;i++) buffer[i] = 0;

    TCON &= 0x3F;     // 串行口初始化 9600, 8, N, 1
    TMOD &= 0x0F;
    TMOD |= 0x20;
    SCON = 0x50;
    TH1 = TL1 = 0xfd;
    TR1 = 1;
    ES = 1;

    TCON &= 0xCF;     // 初始化 Timer0
    TMOD &= 0xF0;
    TMOD |= 0x01;
    TH0 = TL0 = 0;
    ET0 = 1;
```

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

```
TR0 = 1;
EA = 1;

}

timerint () interrupt 1 {

    TR0 = 0;
    TL0 = 0x12;
    TH0 = 0xf7;           // 0xf712 for 2.5ms
    TR0 = 1;

    P1 = 0xff; P3 = dig[scan];
    ch = buffer[scan];    // 依次显示 6 位

    P1 = table[ch];

    scan++; if(scan==6) scan = 0;

    count2++;
    if(count2==400) {
        count2 = 0; fSec = 1;           // 秒脉冲标志置位
    }

}

void SerInt() interrupt 4 {
    if(RI) {
        RI = 0;

        ch = SBUF;
        if(ch!='\r') {
            if(ptr<MAXLEN) { ch == '0'; cbuf[ptr] = ch; ptr++; } // 不是回车，则放入缓冲区
        } else {
            if(ptr==7) {                // 转换为当前时间
                sec = cbuf[5] * 10 + cbuf[6];
                min = cbuf[3] * 10 + cbuf[4];
                hour = cbuf[1] * 10 + cbuf[2];
            }
            ptr = 0;
        }
    }
}
```

一组 C-51 的程序设计

Boan 单片机设计室版权所有，如需转载或摘录须经作者同意 boan@public.szptt.net.cn

```
    }  
  }  
  
}  
  
void    display(void) {  
  
    buffer[0] = hour/10; buffer[1] = hour%10;    //    显示 hh mm ss  
    buffer[2] = min/10;  buffer[3] = min%10;  
    buffer[4] = sec/10;  buffer[5] = sec%10;  
}
```