

单片机与接口技术实验报告

课程名称：单片机	班级：计科 151	
姓名：刘禾子	学号：1500170082	指导教师：张厚武
实验序号：三		实验成绩：

一、实验名称

定时器实现数码管循环显示.

二、实验目的及要求

利用定时器定时对数码管进行循环递增操作

三、实验环境

Keil Uv4

四、实验原理

定时器\计数器工作原理：

定时/计数器实质上是一个加 1 计数器。它随着计数器的输入脉冲进行自加 1，也就是每来一个脉冲，计数器就自动加 1，当加到计数器为全 1 时，再输入一个脉冲就使计数器回零，且计数器的溢出使相应的中断标志位置 1，向 CPU 发出中断请求（定时/计数器中断允许时）。如果定时/计数器工作于定时模式，则表示定时时间已到；如果工作于计数模式，则表示计数值已满。

可见，由溢出时计数器的值减去计数初值才是加 1 计数器的计数值。

工作方式寄存器 TMOD：

工作方式寄存器 TMOD 用于设置定时/计数器的工作方式，低四位用于 T0，高四位用于 T1。其格式如下：

位	7	6	5	4	3	2	1	0	
字节地址：89H	GATE	C/T	M1	M0	GATE	C/T	M1	M0	TMOD

工作方式表：

定时/计数器工作方式设置表

M1M0	工作方式	说 明
00	方式 0	13 位定时/计数器
01	方式 1	16 位定时/计数器
10	方式 2	8 位自动重装定时/计数器
11	方式 3	T0 分成两个独立的 8 位定时/计数器；T1 此方式停止计数

定时器原理：定时/计数器的实质是加 1 计数器（16 位），由高 8 位和低 8 位两个寄存器 THx 和 TLx 组成。TMOD 是定时/计数器的工作方式寄存器，确定工作方式和功能；TCON 是控制寄存器，控制 T0、T1 的启动和停止及设置溢出标志。

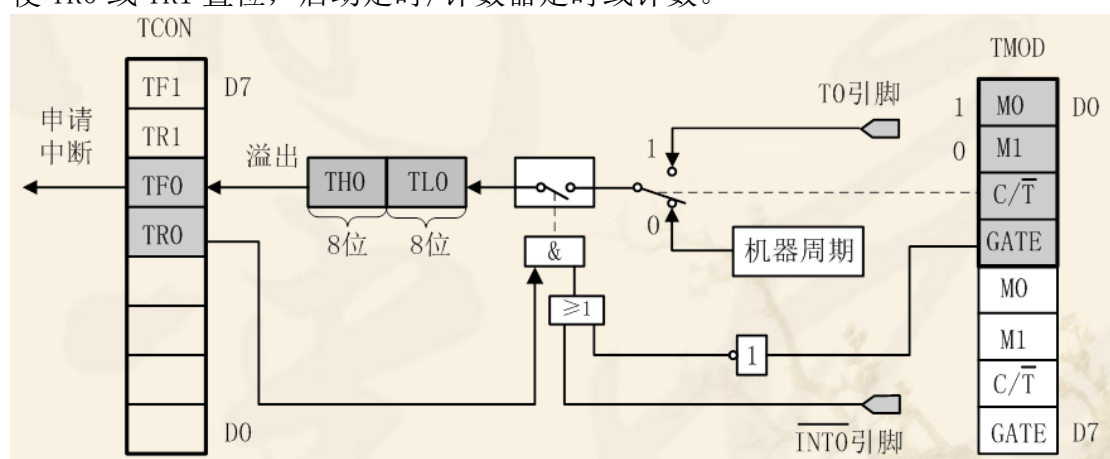
初始化程序应完成如下工作：

对 TMOD 赋值，以确定 T0 和 T1 的工作方式。（本实验采用方式 1，原理图如下）

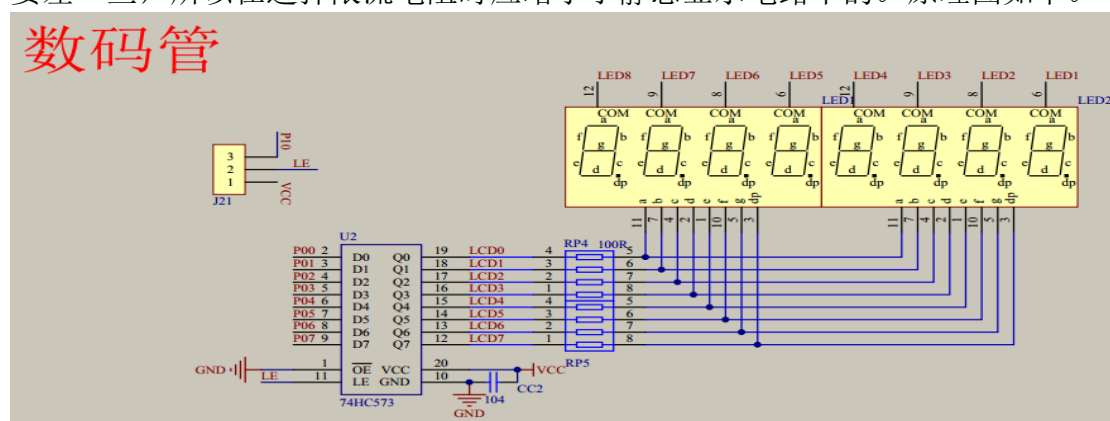
计算初值，并将其写入 TH0、TL0 或 TH1、TL1。

中断方式时，则对 EA 赋值，开放定时器中断。

使 TR0 或 TR1 置位，启动定时/计数器定时或计数。



数码管显示原理：动态显示的特点是将所有数码管的段选线并联在一起，由位选线控制是哪一位数码管有效。选亮数码管采用动态扫描显示。所谓动态扫描显示即轮流向各位数码管送出字形码和相应的位选，利用发光管的余辉和人眼视觉暂留作用，使人的感觉好像各位数码管同时都在显示。动态显示的亮度比静态显示要差一些，所以在选择限流电阻时应略小于静态显示电路中的。原理图如下。



五、实验步骤

程序源码：

```
#include "reg52.h"
#include "intrins.h"
typedef unsigned int u16;      //对数据类型进行声明定义
typedef unsigned char u8;
sbit LSA=P2^2;
sbit LSB=P2^3;
sbit LSC=P2^4;
u8 code smgduan[16]={0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d, 0x07,
                    0x7f, 0x6f, 0x77, 0x7c, 0x39, 0x5e, 0x79, 0x71}; //显示 0~F
的值
u8 a[]={1, 2, 3, 4, 5, 6, 7, 8};
void Timer0() interrupt 1
{
    static u8 k;
    static u8 h;

    TH0=0x3C;    //50000*(11.0592/12MHz)=46080
    TL0=0xB0;
    k++;
    if(k>=10)
    {
        k=0;
        for(h=0;h<8;h++)
        {
            a[h]++;
        }
    }
}
void delay(u16 i)
{
    while(i--);
}
void DigDisplay()
{
    u8 i;
    u8 j;
    for(i=0;i<8;i++)
    {
        P2=_crol_(i, 2);
        j=a[i];
        P0=smgduan[j%9]; //发送段码
    }
}
```

```

        delay(100); //间隔一段时间扫描
        P0=0x00;
    }

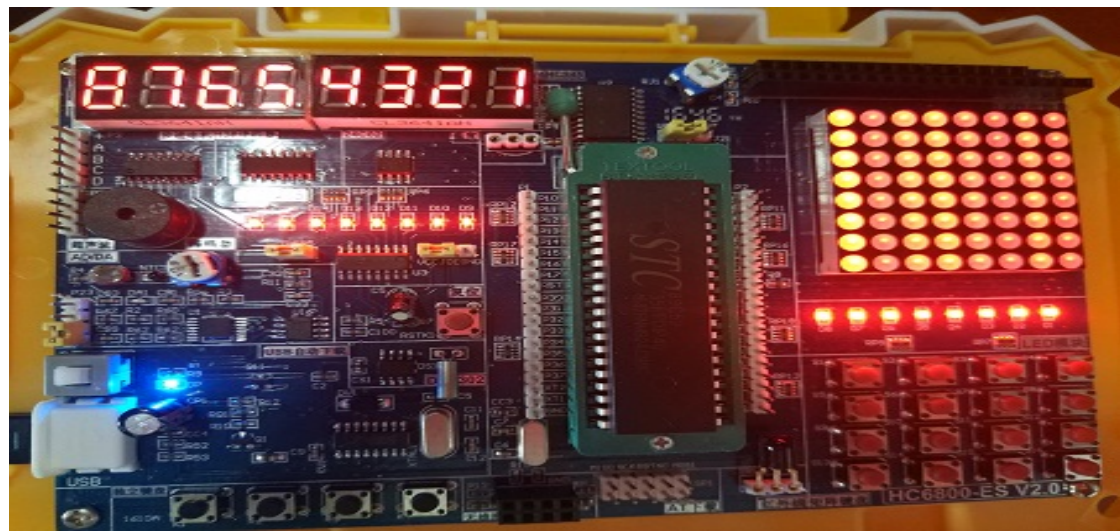
    //消隐}

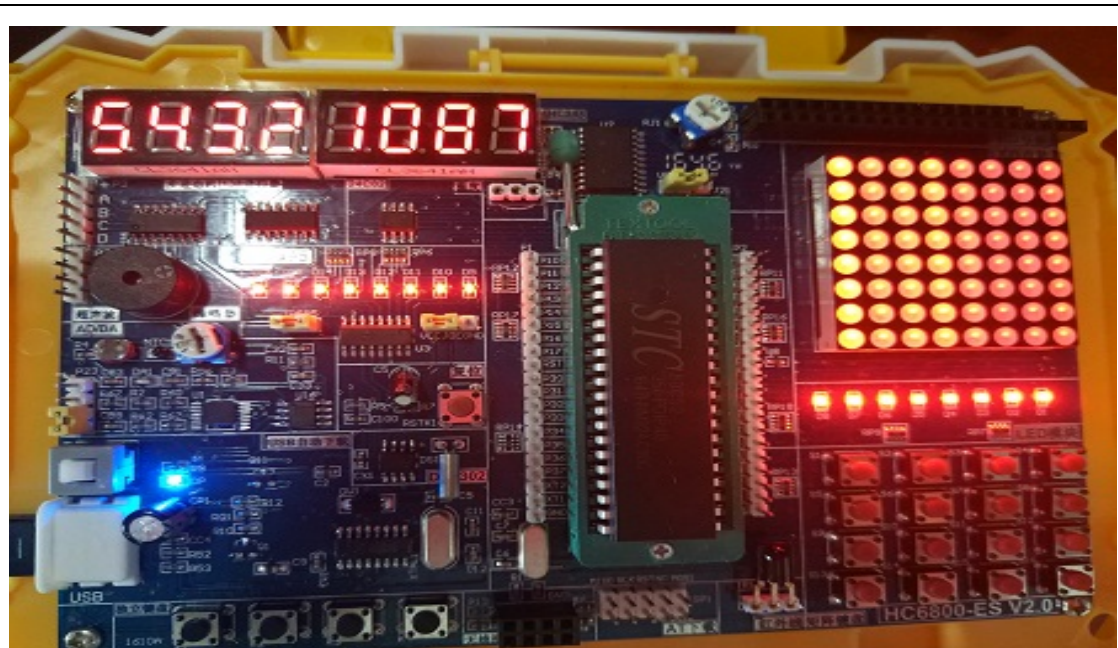
}

void main()
{
    TMOD=0x01;
    TH0=0x3C; //65536-50000*(11.0592/12MHz)
    TL0=0xB0;
    ET0=1;
    EA=1;
    TR0=1;
    while(1)
    {
        DigDisplay();
        //数码管显示函数
    }
}

```

实验结果：





六、总结

通过此次实验对 CPU 的时序有了更深入的认识包括（振荡周期，状态周期，机器周期以及指令周期），再而对定时器初值的各种计算方式有了一定的掌握，最后对于定时器\计数器几种工作方式也有了更深的理解。