
自动化专业综合设计报告

所在实验室： 单片机实验室

指导教师：

学生姓名

班级

学号

成绩评定：

实验一 LED 电子表的设计

一、设计任务要求

(一)设计任务

基于 AT89C51 单片机，制作 LED 显示的智能电子钟。

(二)设计要求

1. 计时，秒，分时，天，月，年。
2. 五路定时输出，可任意关断（最大可到 16 路）
3. 时间，月，日交替显示。
4. 自定义任意时刻自动开关屏。
5. 计时精度：误差 ≤ 1 秒/月。（具有微调设置）
6. 键盘采用动态扫描方式查询。所有的查询，设置功能均由 K1.K2 完成。

二、硬件设计

(一) 总体思路

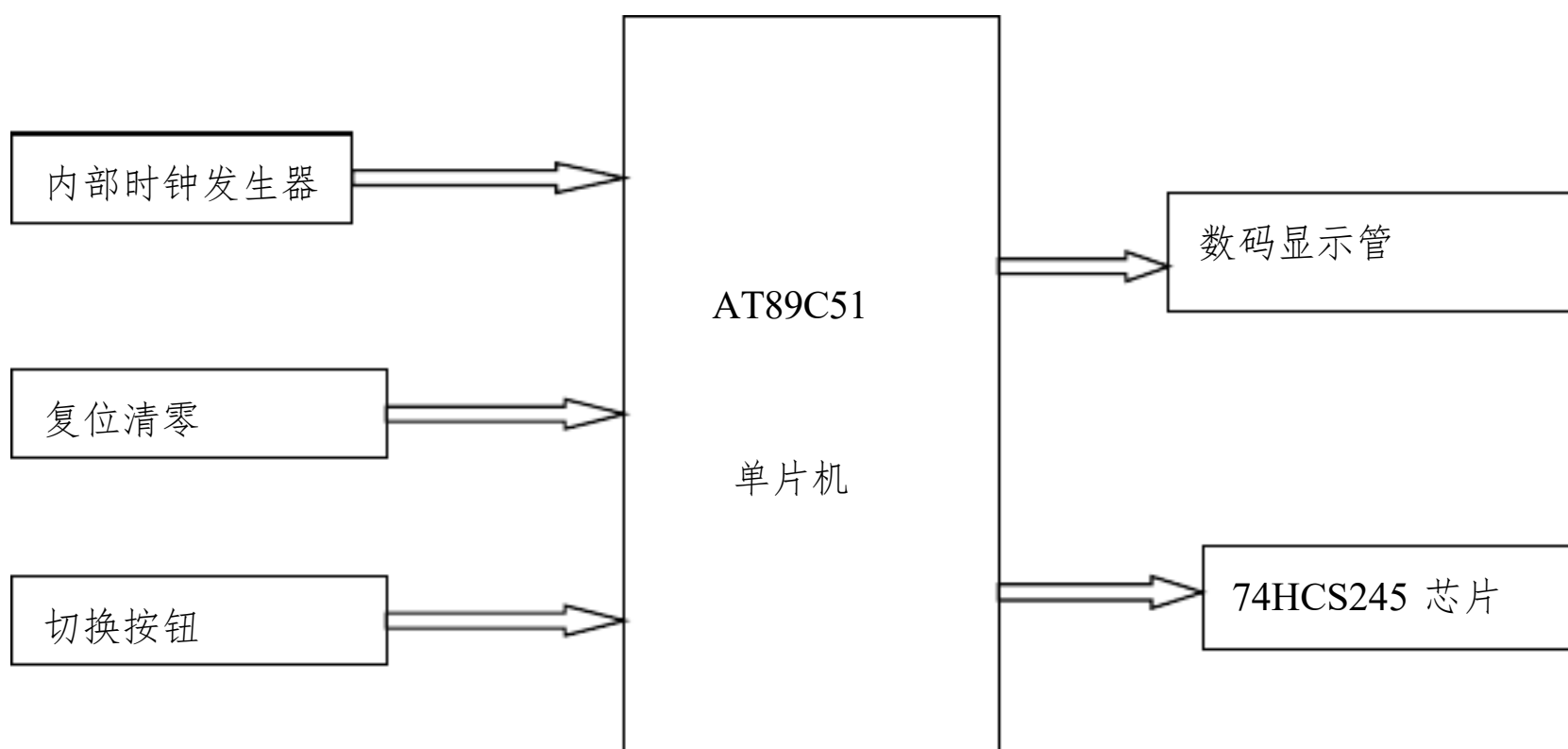


图 1

(二) 实图为

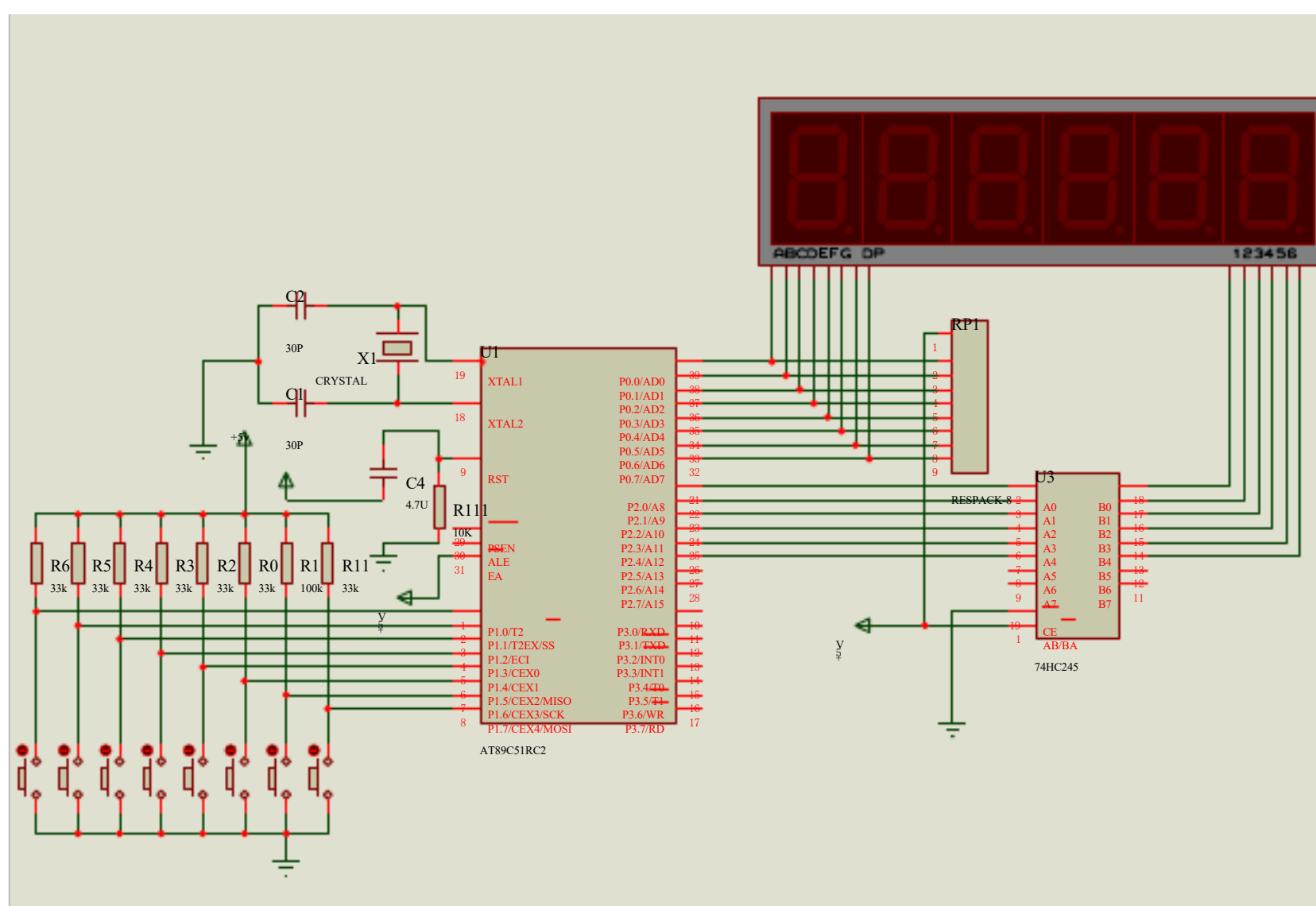


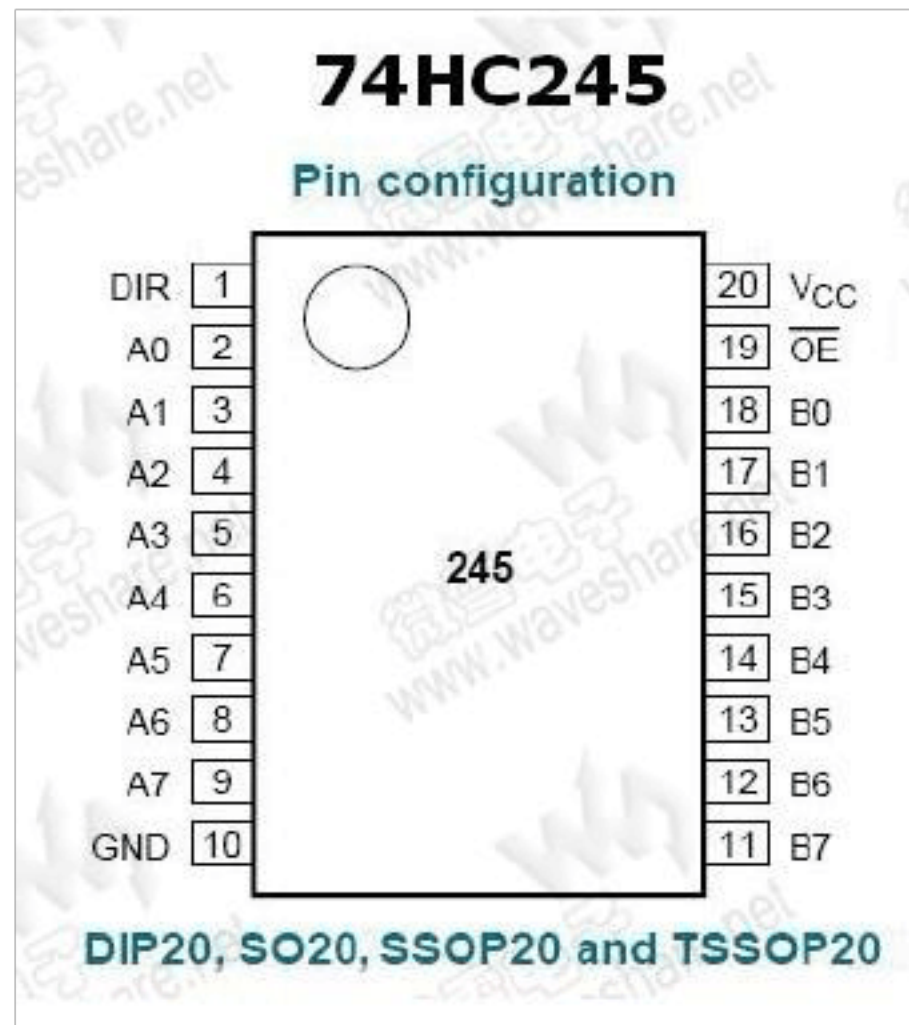
图 2

(三)

1. 首先要选择单片机的芯片，其次，选择系统中要用到的其他外围元件。例如要用到的显示器啊、执行机构等等的。
2. 硬件的设计。硬件设计是指应用系统的电路设计，它包括单片机的芯片、控制电路、存储器、I/O 等等，硬件设计时，应该考虑留有充分的余量，电路设计要保证正确无误，应为在硬件设计中不容易修改设计的结构。
3. 硬件设计时要注意的问题有以下几点：
 - (1) 单片机的选择：尽量选用带有 EEPROM 和 MCS51 系列的单片机，理如我们用到的 AT89C51，这种单片机会减少外部的扩展。
 - (2) 程序存储器：若要外扩存储器时，一般要选用容量较大的 EPROM 芯片，尽量避免用小容量的芯片，应为程序存储容量大些，则可以编程空间充裕。
 - (3) 数据存储器：根据系统功能的要求，如果需要扩展外部 ROM，那么 ROM 芯片可选用 6264 (8KB) 等，扩展外部 ROM 和 RAM 相同，尽量减少芯片的数量。
 - (4) I/O 接口的芯片：I/O 接口的芯片的扩展也需要根据应用系统功能的要求来确定，常用的 I/O 接口芯片一般要选用 8255 芯片，这类芯片可利用功能多、具有口线多、硬件逻辑简单等优点。
 - (5) 总线驱动功能：MCS51 系列的单片机的外部扩展功能很强，但 4 个 8 位并行接口带负载能力是有限的，P0 口能驱动 8 个 TTL 电路，P1~P3 口只能驱动 3 个 TTL 电路。
4. 单片机与显示模块电路

由 74HC245 芯片和电阻，导线构成，控制锁存单片机输出信号。

74HC245 元件的封装：



74HC245 真值表：

TRUTH TABLE 真值表

Control Inputs 控制输入		Operation 运行
$\overline{\text{G}}$	DIR	
L	L	B 数据到A 总线
L	H	A 数据到B 总线
H	X	隔开

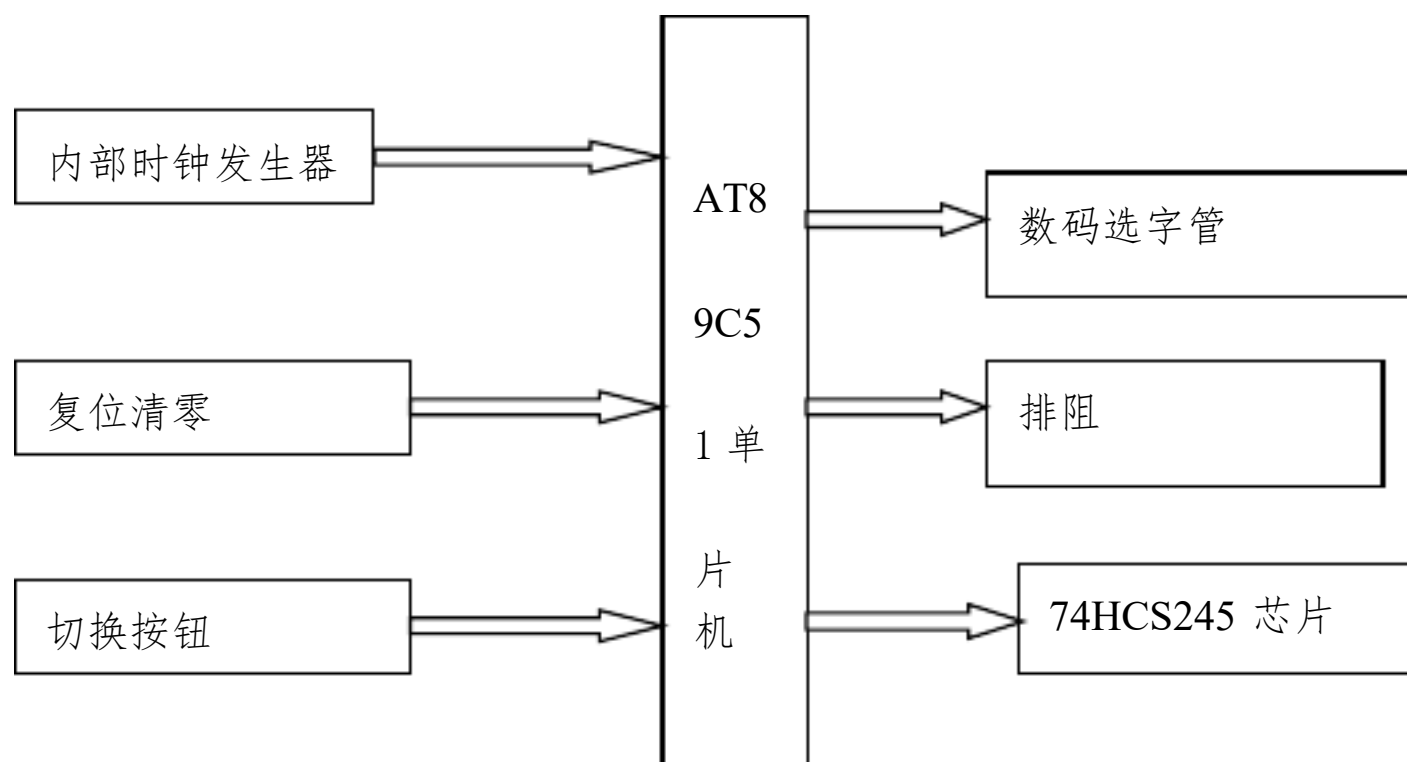
H=高电平 L=低电平 X=不定

5. 显示 LED 电路

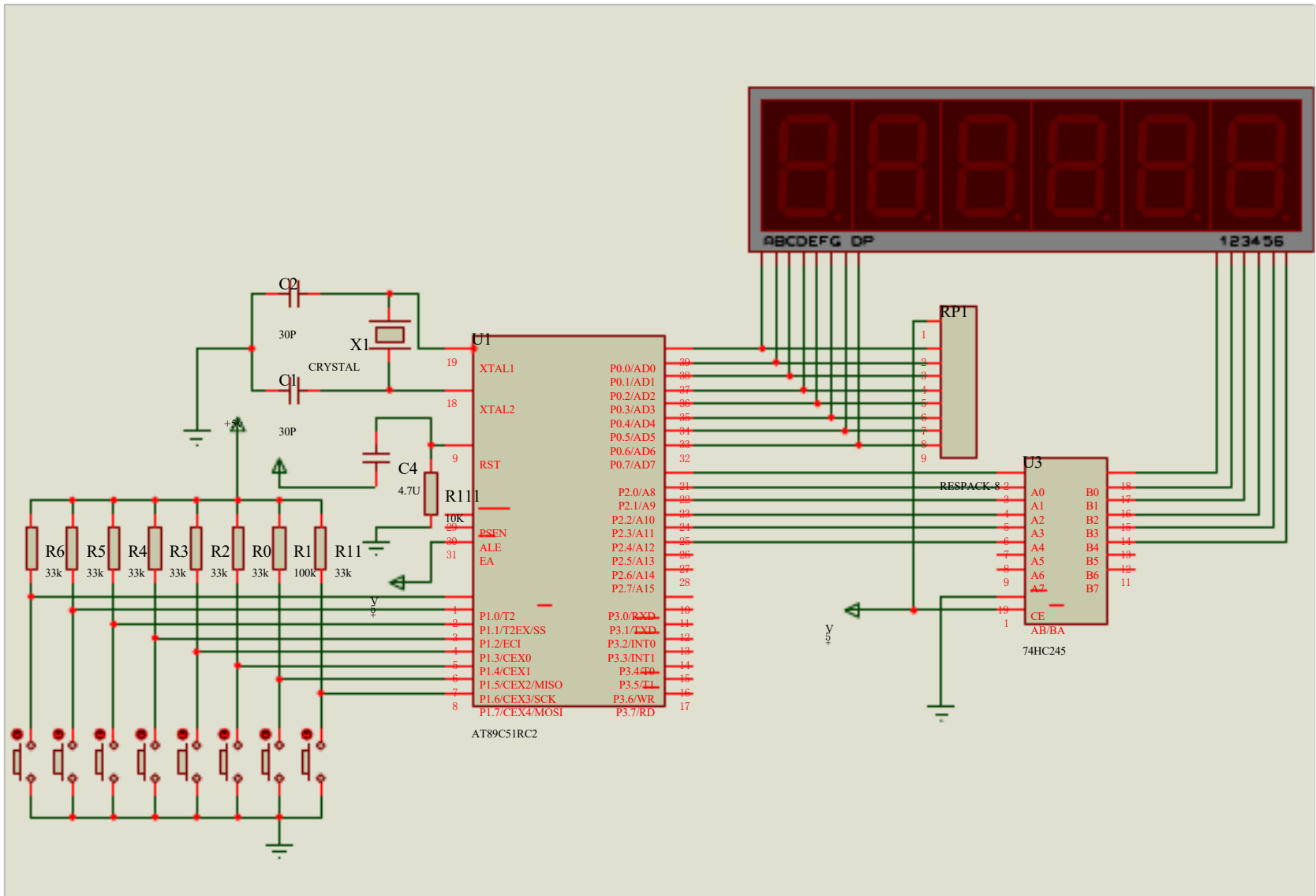
由 7SEG-MPSX6-CC 供阴 LED8 段位显示器构成：

6. 设计的硬件的思路和实图为：

(1) .思路是

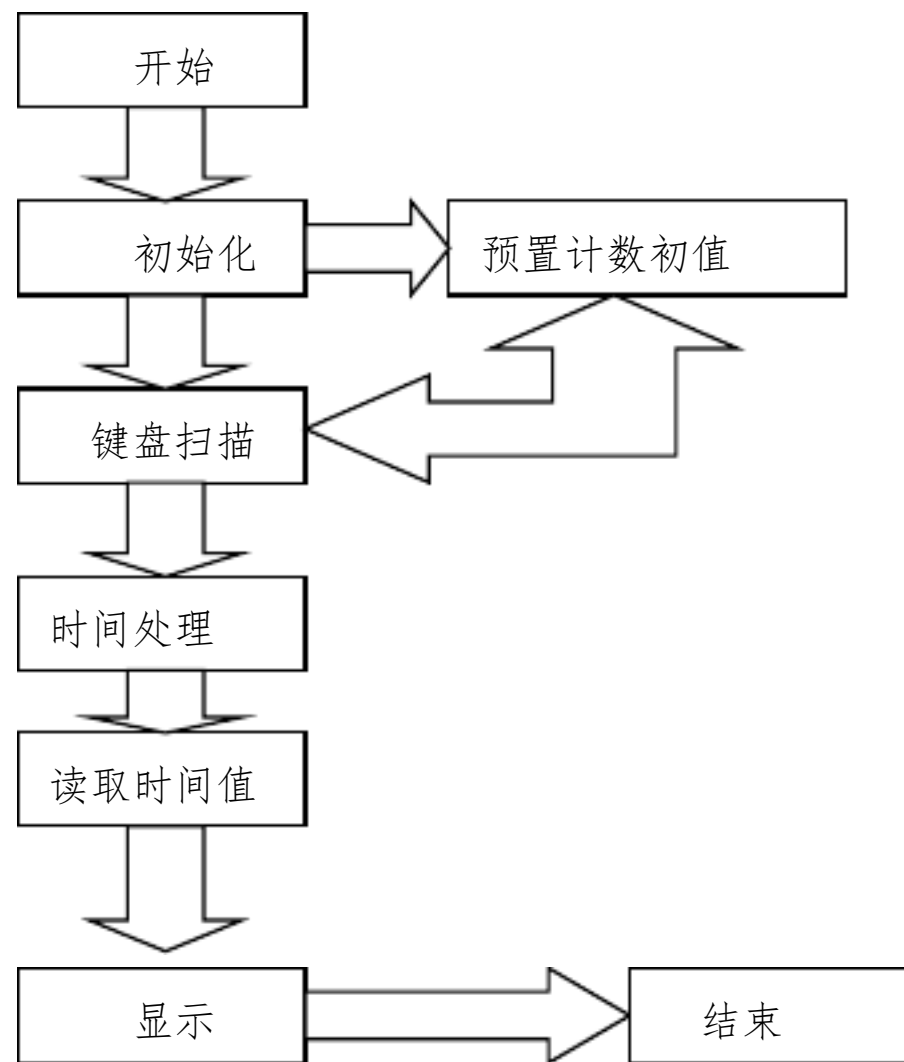


(2) .实图为

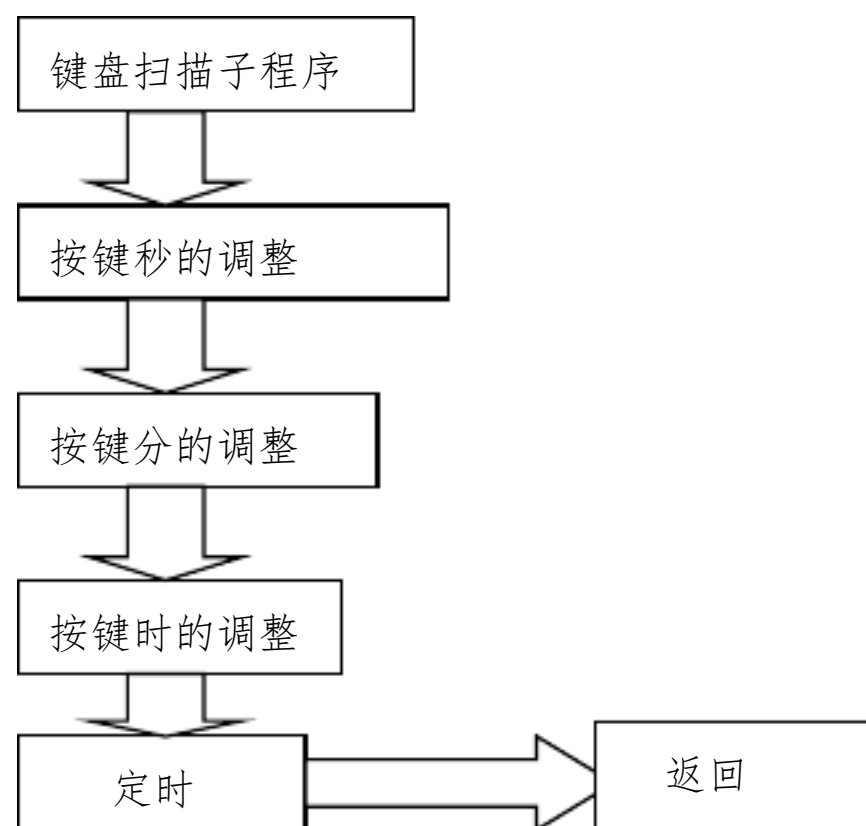


(二) 软件的设计

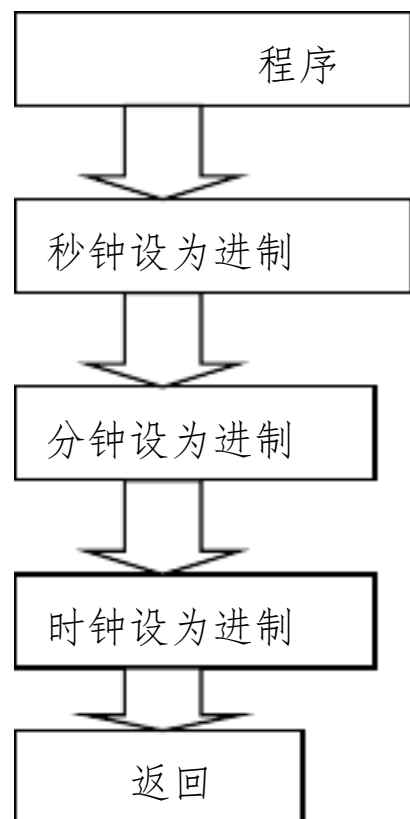
1. 软件的设计，单片机应用系统的软件设计是研究的过程中最关键的一步，没有软件，就无法实现单片机的控制，不同的软件会实现的个功能就有不同的控制，。所以，要编写软件就应该把实现的控制对象及其功能全部掌握，要做到心中有数。
2. 选择了 c 语言来编写程序软件来控制。
3. 流程图为：(1) 主程序流程图



(2) 键盘扫描子程序



3) 时间处理子程序



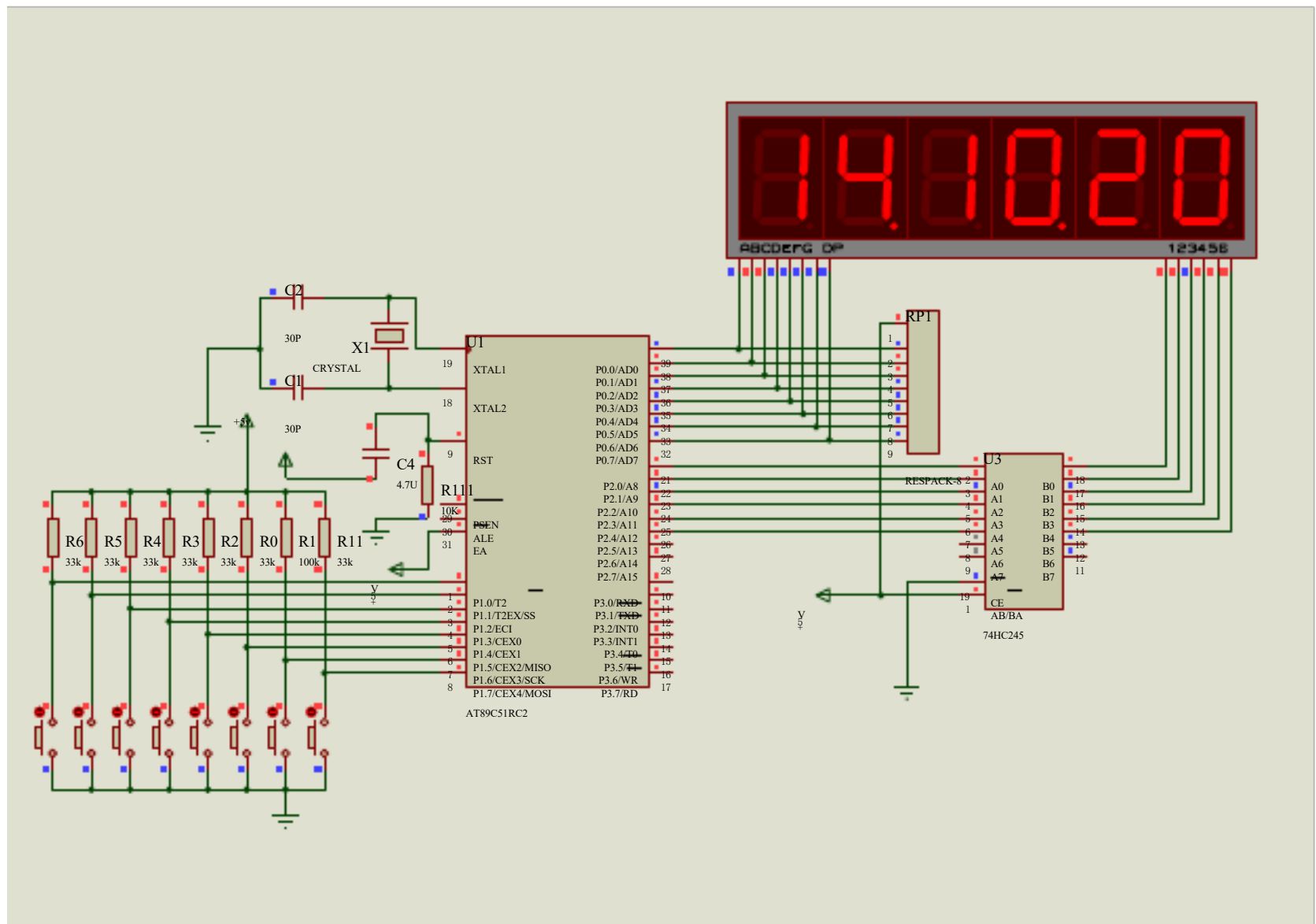
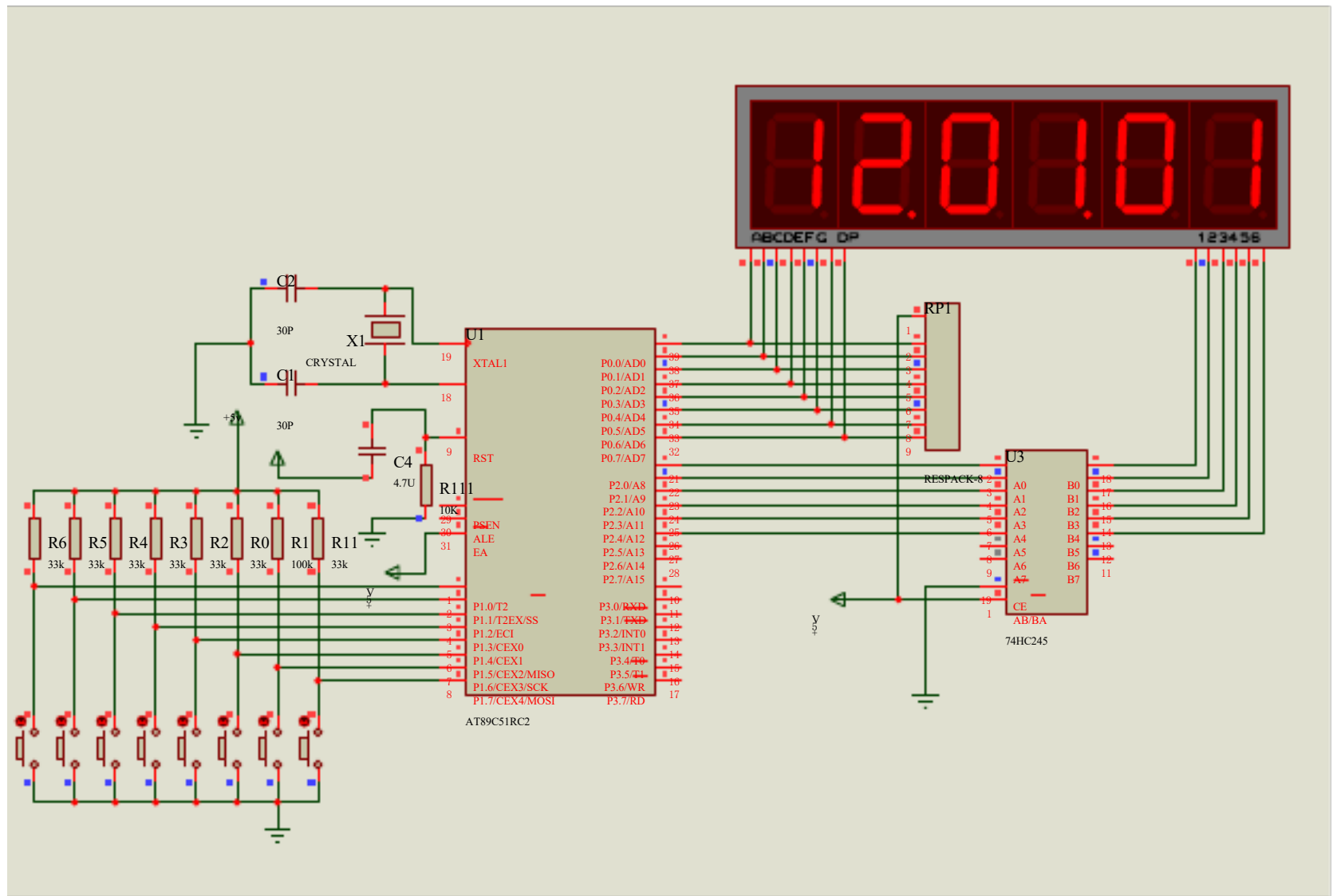
1. 单片机系统的软件千差万别的，没有统一的模式，我们是尽可能的采用模块化的结构，并按照先粗后细的方法，把整个系统软件分成多个独立模块。应该确定个模块的结构和功能，个模块的接口信息。
2. 各个模块可以分别独立设计，编辑和调试，最后将各个模块连接成一个完整的程序进行调试，。
3. 较为复杂的软件是建立在各个机本末快的基础上的。
4. 系统的测试：系统的测试包括硬件调适合软解调试，硬件的调试是解决硬件的故障，包括设计故障和工艺错误。软件的调试是经过在线的仿真调试，出了发现和解决程序的错误外，也可已发现硬件的错误。
5. 程序调试是按模块进行的，各个模块和子程序都是独立进行的，最后是要连在一起来调试的，可以利用开发工具的单步和断点运行方式，通过检查系统的 CPU 现场、RAM 和 SFR 的内容及 I/O 口的状态，来检查程序大的执行结果和系统 I/O 设备

误。

在调试的过程中要不断的调整、修改系统的软件和硬件，直到其符合预期结果为止，

联机调试运行后，将软件固化到 EPROM 中。

三、硬件设计的电路和调试结果为



首先要选择单片机的芯片，其次，选择系统中要用到的其他外围元件。例如，我们要用到的显示器啊、执行机构等等的。

硬件的设计。硬件设计是指应用系统的电路设计，它包括单片机的芯片、控制电路、存储器、I/O 等等，硬件设计时，应该考虑留有充分的余量，电路设计要保证正确无误，应为在硬件设计中不容易修改设计的结构。

硬件设计时要注意的问题有以下几点：（1）单片机的选择：尽量选用带有 EEPROM 和 MCS51 系列的单片机，理如我们用到的 AT89C51，这种单片机会减少外部的扩展。

（2）程序存储器：若要外扩存储器时，一般要选用容量较大的 EPROM 芯片，尽量避免用小容量的芯片，应为程序存储容量大些，则可以编程空间充裕。（3）数据存储器：

根据系统功能的要求，如果需要扩展外部 ROM，那么 ROM 芯片可选用 6264（8KB）

等，扩展外部 ROM 和 RAM 相同，尽量减少芯片的数量。（4）I/O 接口的芯片：I/O

接口的芯片的扩展也需要根据应用系统功能的要求来确定，常用的 I/O 接口芯片一般

要选用 8255 芯片，这类芯片可利用功能多、具有口线多、硬件逻辑简单等优点。（5）

总线驱动功能：MCS51 系列的单片机的外部扩展功能很强，但 4 个 8 位并行接口带负

载能力是有限的，P0 口能驱动 8 个 TTL 电路，P1~P3 口只能驱动 3 个 TTL 电路。

```
/******  
/*延时子程序*/  
/******  
void delay (uchar k)  
{  
uchar j;  
while ((k--)!=0)  
{  
for (j=0;j<125;j++)  
{;
```

```
    }  
}  
/*****/  
/*时间处理子程序*/  
/*****/  
void time_pro( void )  
{  
    if(second==60)                //秒钟设为进制  
    {  
        second=0;  
        minute++;  
        if( minute==60)          //分钟设为进制  
        {  
            minute=0;  
            hour++;  
            if(hour==24)         //时钟设为进制  
            {  
                hour=0;  
                day++;  
                // switch(mon)  
                // {  
                //  
                // case 3:  
                // case 5:  
                // case 7:  
                // case 8:  
                // case 10:  
                // case 12:  
                // case 1:  
                if(day == 30)  
                {  
                    day=1;  
                    mon++;  
                // }  
                // break;  
                // case 2:  
                // if(day ==31)  
                // {  
                //     day=1;  
                //     mon++;  
                // }  
                // break;  
            }  
        }  
    }  
}
```

```

        P0=(dispcode[minute%10])|0X80;           //显示分的个位
        delay(4);
        P2=0xef;
        P0=dispcode[second/10];                 //显示秒的十位
        delay(4);
        P2=0xdf;
        P0=dispcode[second%10];                 //显示秒的个位
        delay(4);
    }
    /*年月日显示程序*/
    void display1( void )
    {
        P2=0xfe;
        P0=dispcode[year/10];
        delay(4);
        P2=0xfd;
        P0=(dispcode[(year%10)])|0X80;
        delay(4);
        P2=0xfb;
        P0=dispcode[mon/10];
        delay(4);
        P2=0xf7;
        P0=(dispcode[mon%10])|0X80;
        delay(4);
        P2=0xef;
        P0=dispcode[day/10];
        delay(4);
        P2=0xdf;
        P0=dispcode[day%10];
        delay(4);
    }

    /*****/
    /*主函数*/
    /*****/
    void main( void )
    {
        P1=0xff;           //初始化p1口，全设为
        TMOD = 0x11; //time0 为定时器，方式
        TH0=0x3c;         //预置计数初值
        TL0=0xb0;
        EA=1;
        ET0=1;
        TR0=1;
        while (1)

```

```
{
    keyscan(); //按键扫描
    time_pro(); //时间处理
    if(disp)
        display1(); //显示时间
    else
        display();
}
}
```

五、调试过程和方法

1. 单片机系统的软件千差万别的，没有统一的模式，尽可能的采用模块化的结构，并按照先粗后细的方法，把整个系统软件分成多个独立模块。应该确定每个模块的结构和功能，每个模块的接口信息。

各个模块可以分别独立设计，编辑和调试，最后将各个模块连接成一个完整的程序进行调试。

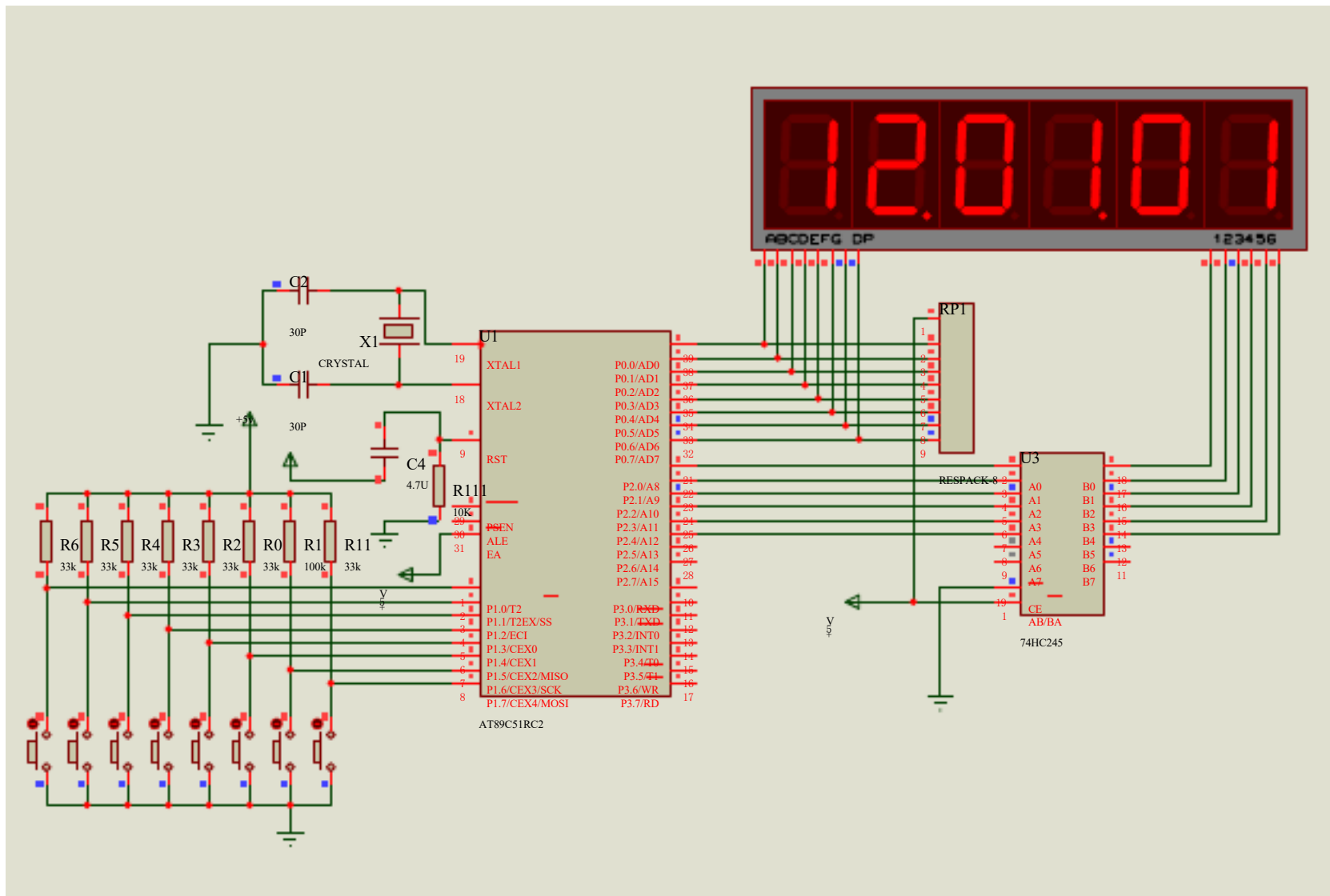
2. 系统的测试包括硬件调试和软件调试，硬件的调试是解决硬件的故障，包括设计故障和工艺错误。软件的调试是经过在线的仿真调试，除了发现和解决程序的错误外，也可能已发现硬件的错误。

3. 程序调试是按模块进行的，各个模块和子程序都是独立进行的，最后是要连在一起调试的，可以利用开发工具的单步和断点运行方式，通过检查系统的 CPU 现场、RAM 和 SFR 的内容及 I/O 口的状态，来检查程序的执行结果和系统 I/O 设备的状态是否正常，从中发现程序的逻辑错误、转移地址的错误以及随机的录入错误。

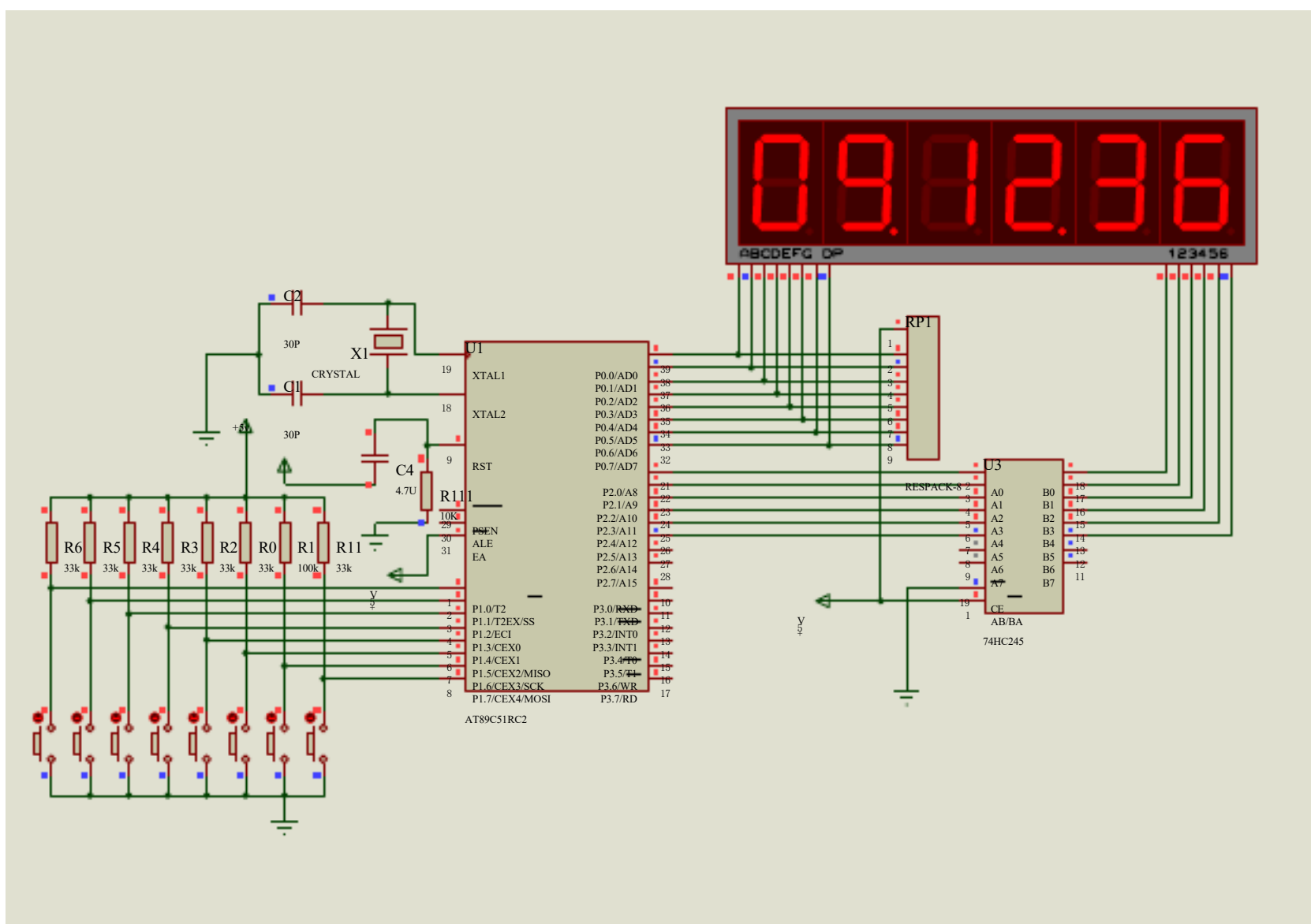
4. 在调试的过程中要不断的调整、修改系统的软件和硬件，直到其符合预期结果为止，

联机调试运行后，将软件固化到 EPROM 中。

1. 这是日期的调试图：分别为年、月、日



2. 此图为时、分、秒的调试图



以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/007104020050006150>