



杭州电子科技大学  
HANGZHOU DIANZI UNIVERSITY

## 《单片机原理及应用》作业报告

### 实验报告 4 第三部分：24 秒倒计时

学院 卓越学院

学号 23040447

姓名 陈文轩

专业 智能硬件与系统(电子信息工程)

2025 年 5 月 12 日

设计实现 24 秒计时器。要求：(1) 倒计时功能 (2) 24 秒复位键功能 (3) 启动/暂停键功能

## 1 实验代码

Code Listing 1: 实验程序

```
1  #include <reg51.h> // 单片机头文件
2
3  unsigned char code Tab[] = {0xC0, 0xF9, 0xA4, 0xB0, 0x99, 0x92, 0x82, 0xF8,
    0x80, 0x90}; // 共阳数码管码段表
4  unsigned char Dat[] = {0, 0, 2, 4}; // 初始时间为24秒
5  int Dat_int=24;
6  int i; // 定义变量，作为循环
7  unsigned char tmp; // 定义片选变量
8  unsigned char KeyState = 0x0F; // 按键状态变量，初始值为高电平（未按下）
9  bit isRunning = 1; // 计时器运行状态标志位
10
11 #define KEY1 (P1 & 0x01) // 按键1连接到 P1.0
12 #define KEY2 (P1 & 0x02) // 按键2连接到 P1.1
13 #define KEY3 (P1 & 0x04) // 按键3连接到 P1.2
14 #define KEY4 (P1 & 0x08) // 按键4连接到 P1.3
15 #define TIM_SET 24 //此程序设置几秒定时 方便调试
16
17 void Delay() // 延时子程序，作为数码管显示延迟
18 {
19     unsigned char i;
20     for (i = 0; i < 100; i++);
21 }
22
23 void Timer1_Init() // 定时器1初始化
24 {
25     TMOD |= 0x10; // 设置定时器1为模式1（16位定时器）
26     TH1 = 0x00; // 定时器初值高字节（1秒定时）
27     TL1 = 0x00; // 定时器初值低字节
28     ET1 = 1; // 使能定时器1中断
29     EA = 1; // 使能总中断
30     TR1 = 1; // 启动定时器1
31 }
32
33 void Timer1_ISR() interrupt 3 // 定时器1中断服务函数
34 {
```

```
35     static int counter=0;
36     TH1 = 0x3C;    // 重装定时器初值高字节
37     TL1 = 0xAF;    // 重装定时器初值低字节 (15535)D=(3CAF)H 即 65535
                     -50000=15535 延时50ms;
38
39     counter++;
40     if (counter==20) //50*20=1000ms 1s 定时器中断
41     {
42         counter=0;
43         Dat_int--;
44         if (Dat_int<0)
45         {
46             Dat_int=TIM_SET;
47         }
48         Dat[2]=Dat_int/10;
49         Dat[3]=Dat_int%10;
50     }
51 }
52
53 void Ext0_Init() // 外部中断0初始化
54 {
55     ITO = 1; // 设置外部中断0为下降沿触发
56     EX0 = 1; // 使能外部中断0
57     EA = 1;  // 使能总中断
58 }
59
60 void Ext0_ISR() interrupt 0 // 外部中断0服务函数 正常工作
61 {
62     //进入中断后,说明有按键按下,再进行扫描,这样大大节省程序运行资源
63     Delay(); //延时消抖
64     if (KEY1==0) //按键1 p1.0 按下 实现复位
65     {
66         Dat_int=0;
67     }
68     if (KEY2==0)
69     {
70         TR1=(TR1==1)?0:1; //切换TRI状态 控制定时器1开启与否
71     }
72 }
73 }
74
75 void main()
76 {
```

```

77 Timer1_Init(); // 初始化定时器1
78 Ext0_Init();   // 初始化外部中断0
79
80 Dat[2]=TIM_SET/10;
81 Dat[3]=TIM_SET%10; //按照宏定义初始化Dat(数码管显示直接控制数组)
82 Dat_int=TIM_SET;
83
84
85 while (1) // 无限循环 数码管扫描主进程
86 {
87     tmp = 0x01; // 片选初值
88     for (i = 0; i < 4; i++) // 循环4次
89     {
90         P2 = tmp; // 片选初值
91         P0 = Tab[Dat[i]]; // 输出某一位数字的码段值
92         tmp = tmp << 1; // 片选值左移一位
93         Delay(); // 调用延时
94     }
95 }
96 }

```

## 2 实验效果

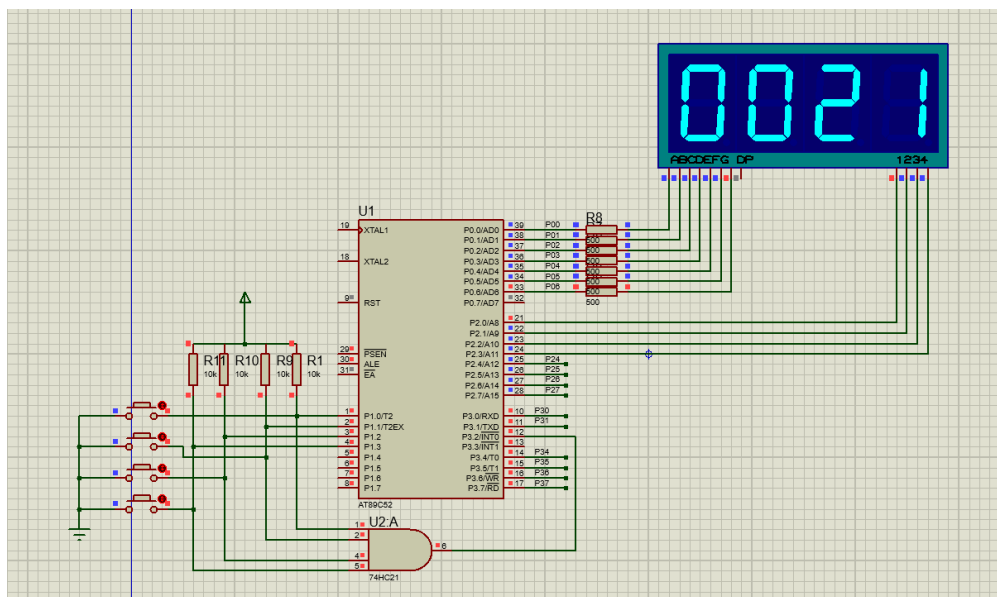


图 1 正常计时：显示到 21

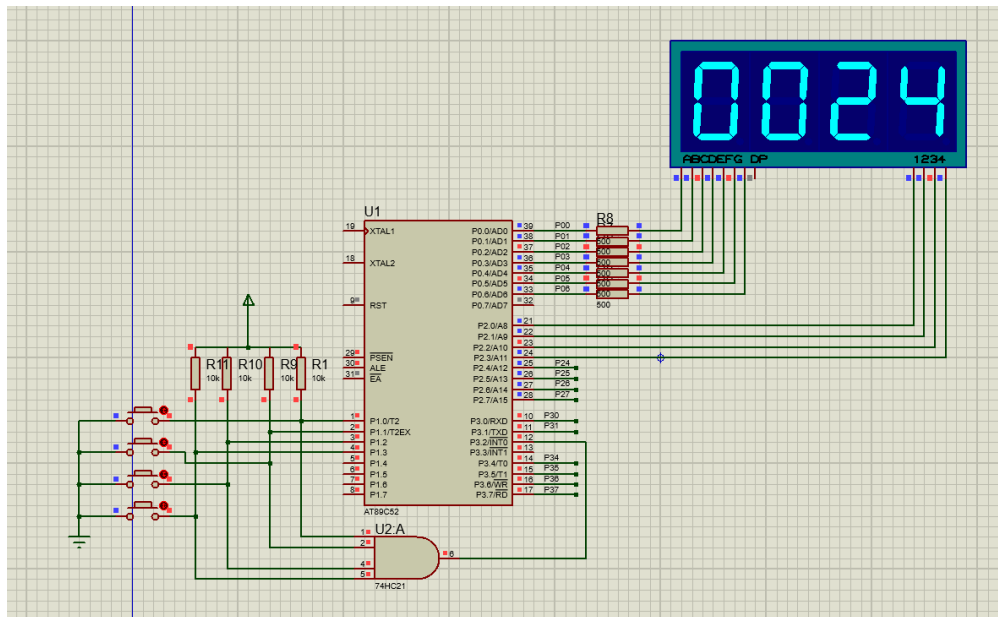


图 2 复位至 24

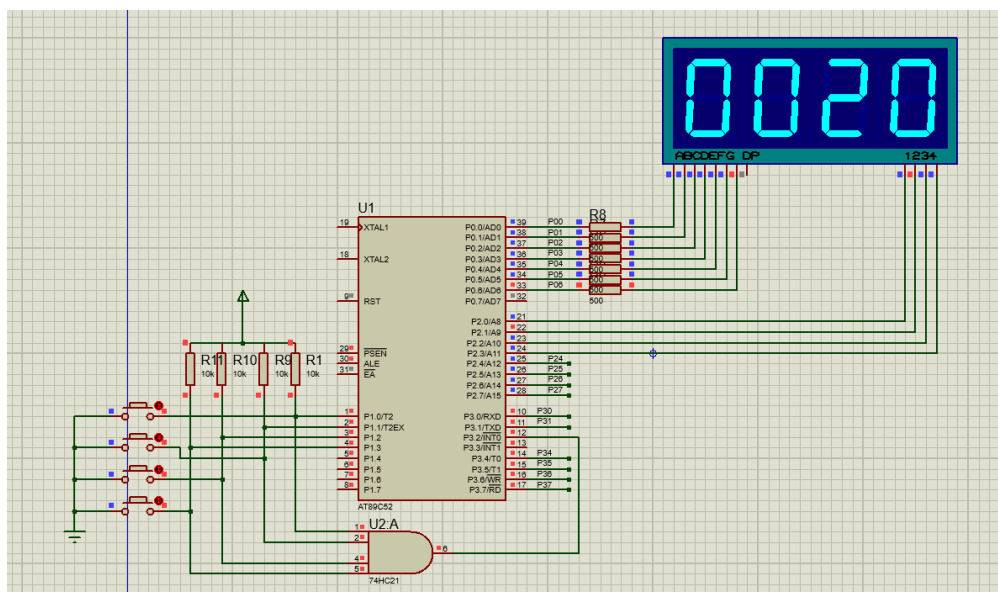


图 3 暂停在 20

### 3 流程图

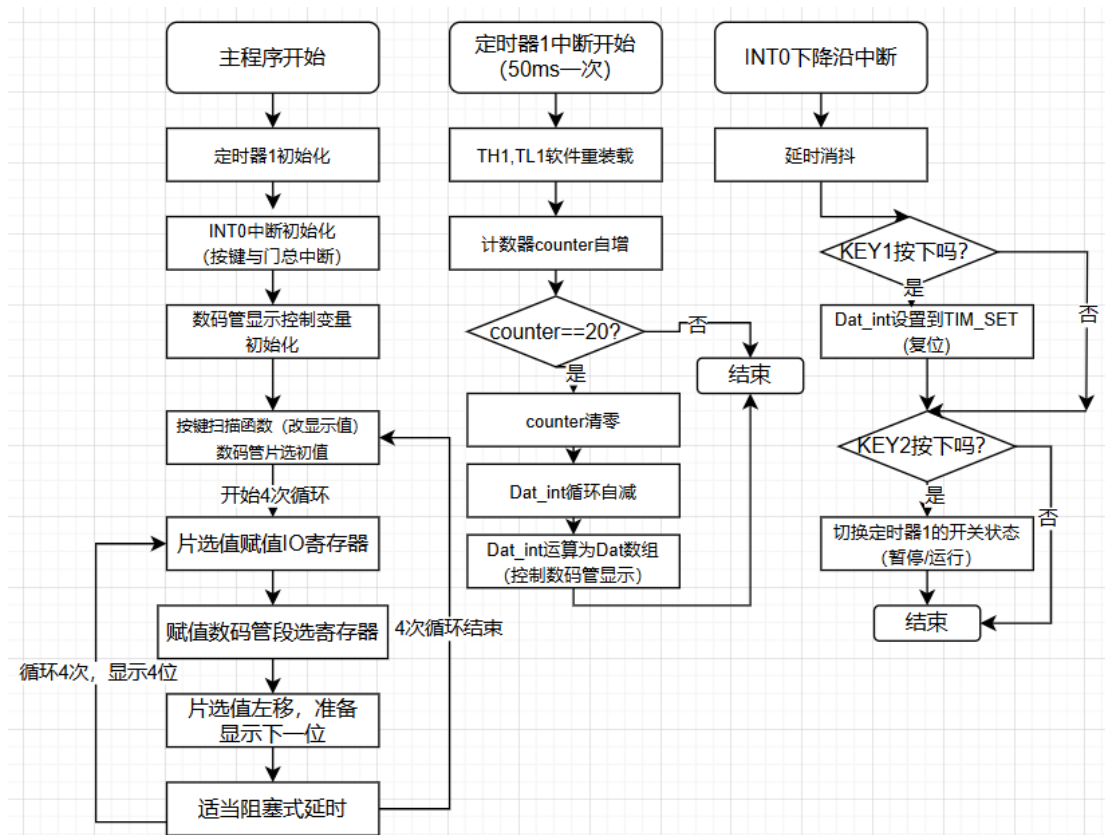


图 4 系统控制流程图

### 4 实验体会

本实验通过动态扫描方式结合四输入与门的中断控制，实现按键的中断触发扫描，以此作为按键实现相应功能之基础；结合定时器溢出中断，实现定时显示值自减；在迁移已有的数码管主线程显示，实现系统的整体功能；同时也发现定时不一定准确（需要调参），暂停时复位无相应及时显示等问题，需要做出一定改进。