
项目六 独立按键的控制

陈超然

汕头职业技术学院

目录

1. 认识键盘

- 键盘的应用领域
- 键盘的分类
- 常见的非编码键盘

2. 独立按键控制数码管加一显示

- 独立按键与单片机的接口电路
- 检测独立按键的程序设计

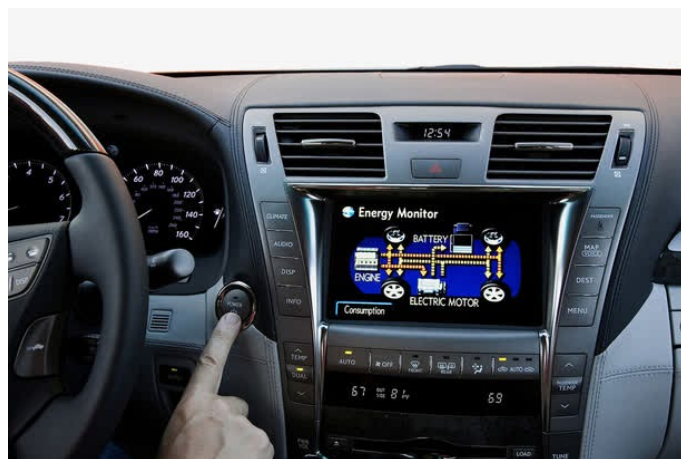
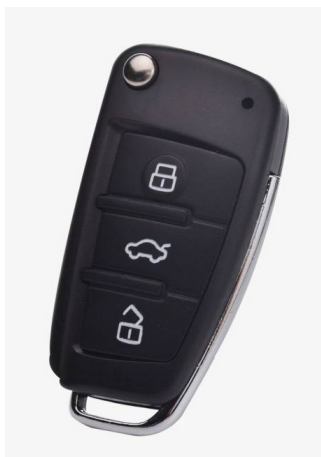
3. 思考与改进

- 独立按键的结构
- 按键抖动现象
- 软件消抖的原理和程序设计

4. 实战作业

6.1 认识键盘

1、应用领域



6.1 认识键盘

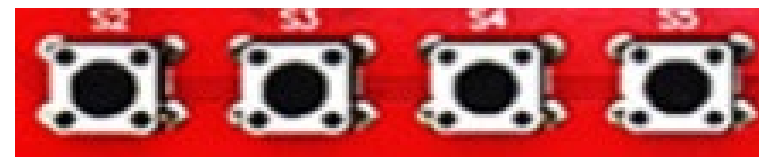
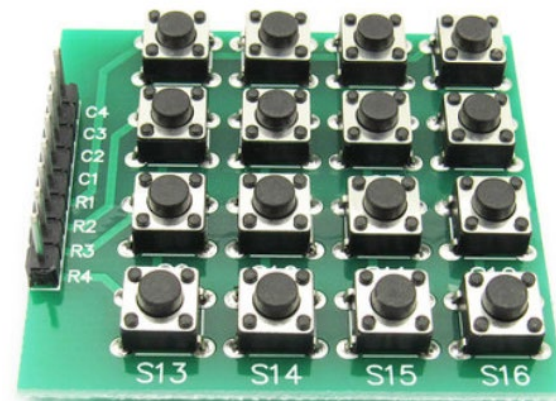
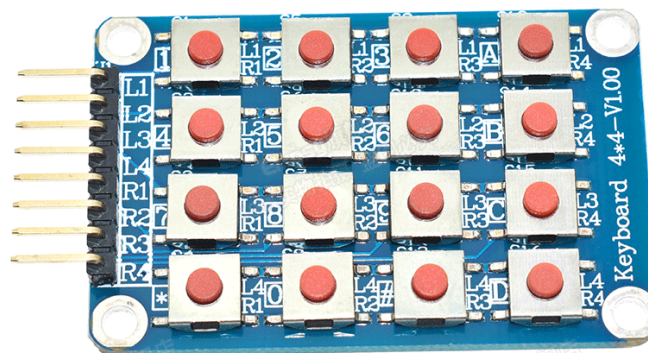
2、键盘的分类

- 键盘分为**编码键盘**和**非编码键盘**。
 - 编码键盘：键盘上闭合键的识别由专用的硬件编码器实现，并产生键编码号或键值，如计算机键盘。
 - 非编码键盘：靠软件编程来识别的键盘，在单片机组成的各种系统中，用的较多的是非编码键盘。
- 因此，在单片机中，我们主要讲非编码键盘。
- 非编码键盘又分为**独立按键**和**矩阵键盘**。

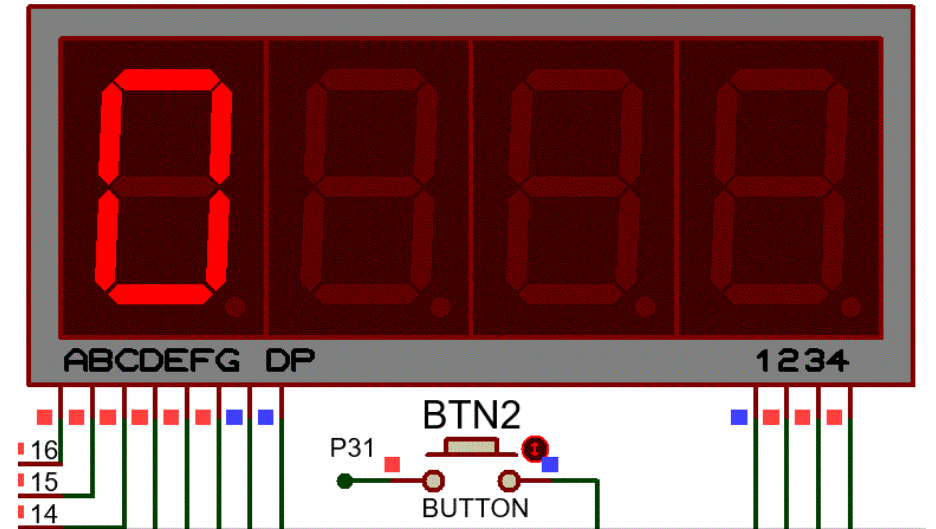
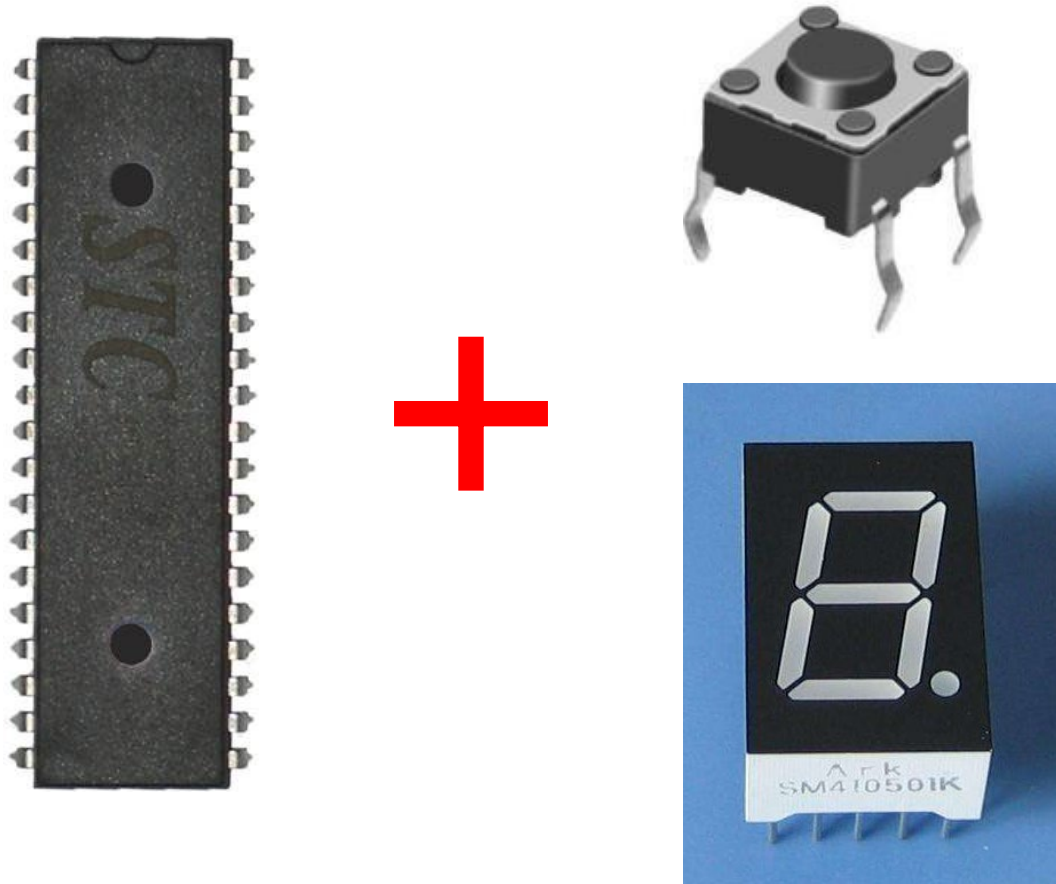


6.1 认识键盘

3、常见的非编码键盘



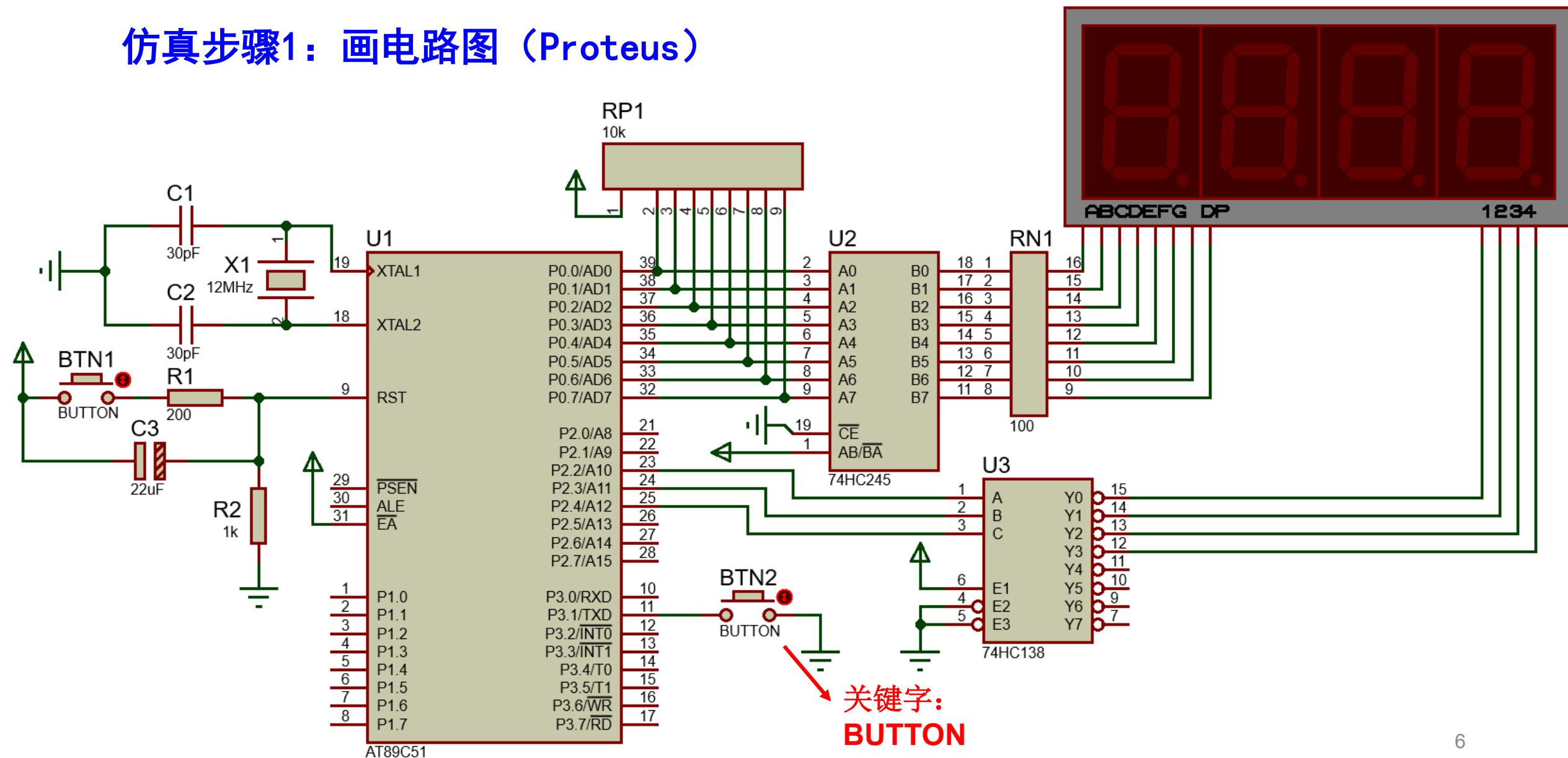
6.2 独立按键控制数码管加一显示



效果展示

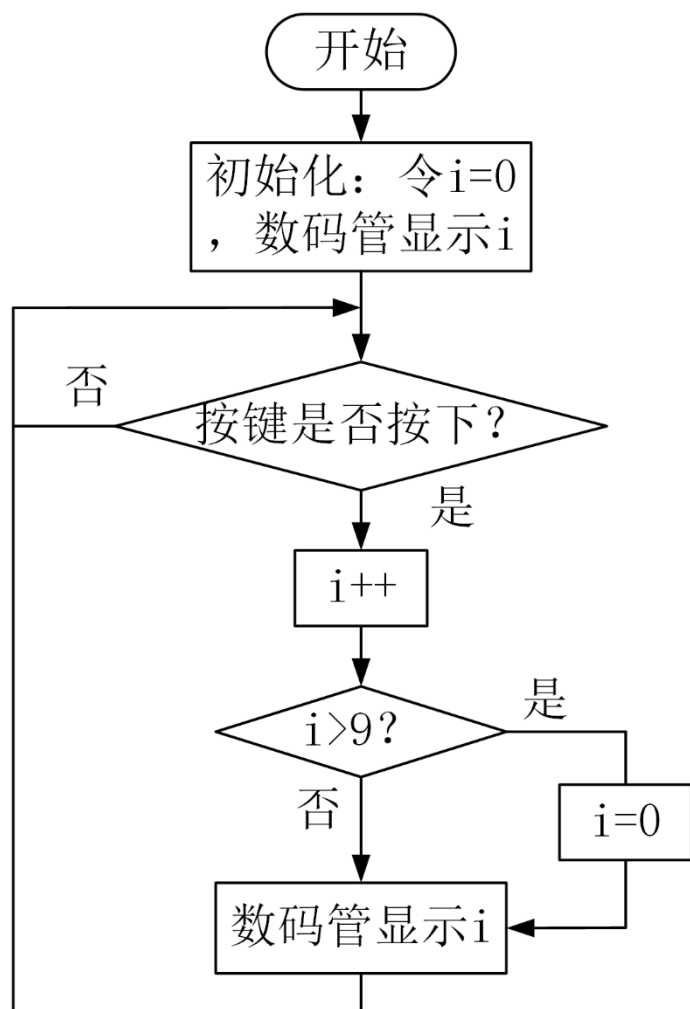
6.2 独立按键控制数码管加一显示

仿真步骤1：画电路图（Proteus）



6.2 独立按键控制数码管加一显示

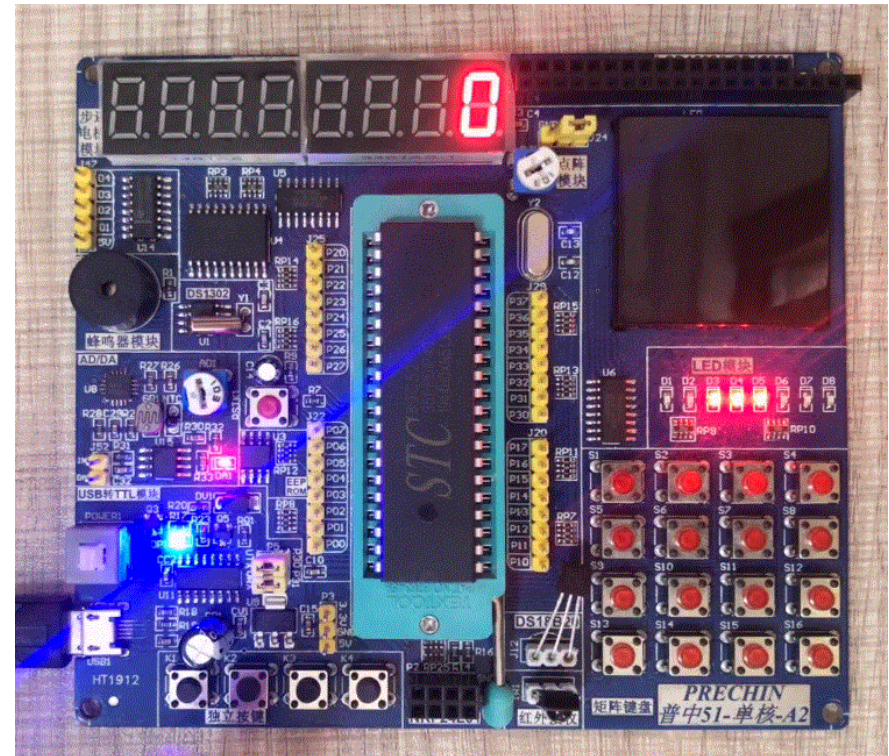
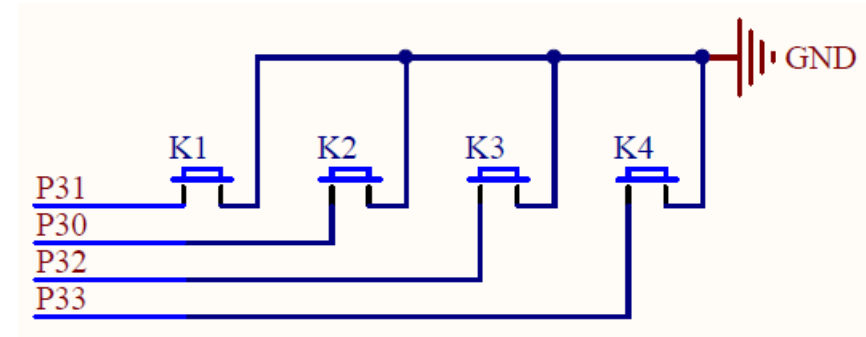
仿真步骤2：画流程图 & 编程（Keil）



```
#include <regx51.h>
#define uint unsigned int
#define uchar unsigned char
sbit key=P3^1;
uchar code table[]={0x3f,0x06,0x5b,
0x4f,0x66,0x6d,0x7d,0x07,0x7f,0x6f,
0x77,0x7c,0x39,0x5e,0x79,0x71};
void main()
{
    uchar num=0;
    P2=0xe3;
    P0=table[0];
    key=1; //读引脚前置高电平
    while(1)
    {
        if(key==0)
        {
            num++;
            if(num>9)
                num=0;
            while(key==0); //松开按键才加1
            P0=table[num];
        }
    }
}
```

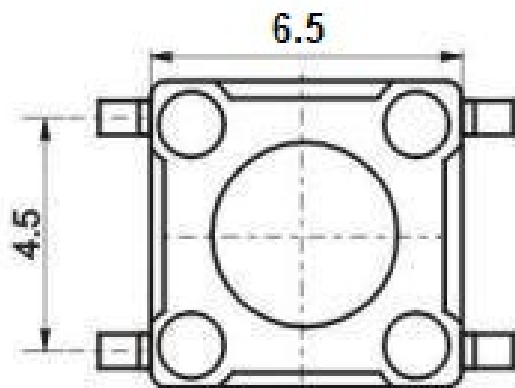

6.3 思考与改进

- Proteus仿真似乎没什么问题，但把程序下载到实际开发板呢？
- 实际开发板情况：
每按一次按键，数码管不一定加1显示：
有时加1，有时加2，有时加3等等
- 思考：什么原因导致Proteus仿真与实际开发板不同？

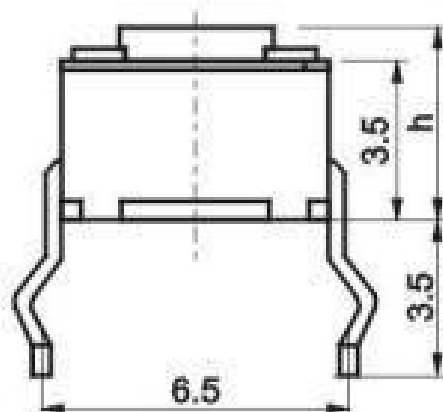
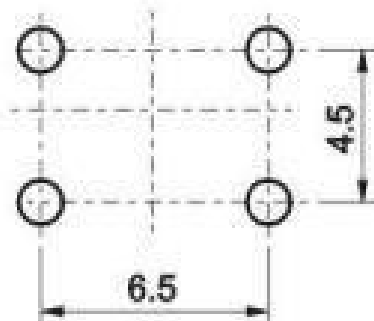


6.3 思考与改进

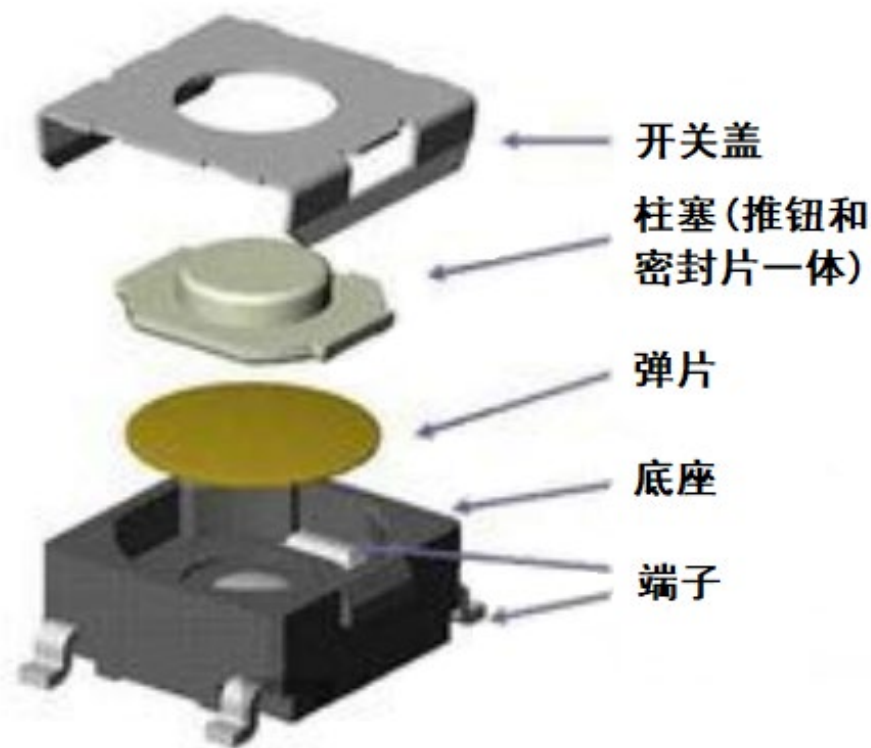
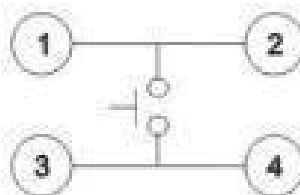
独立按键的结构



P.C.B.Land Pattern

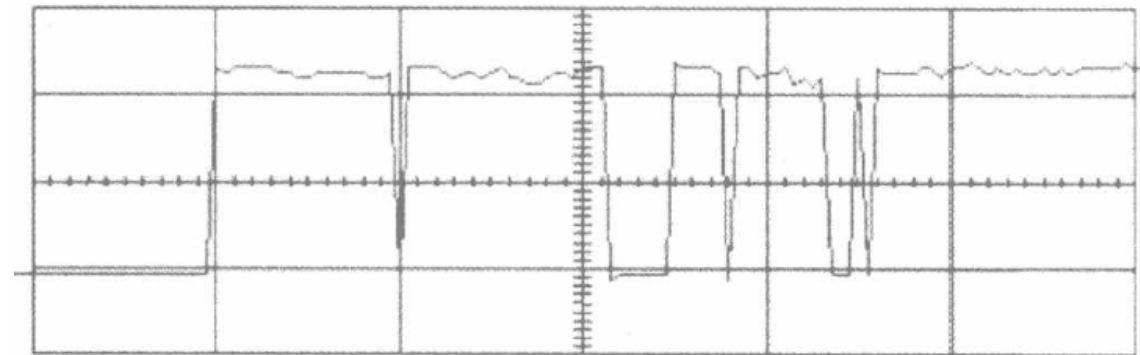
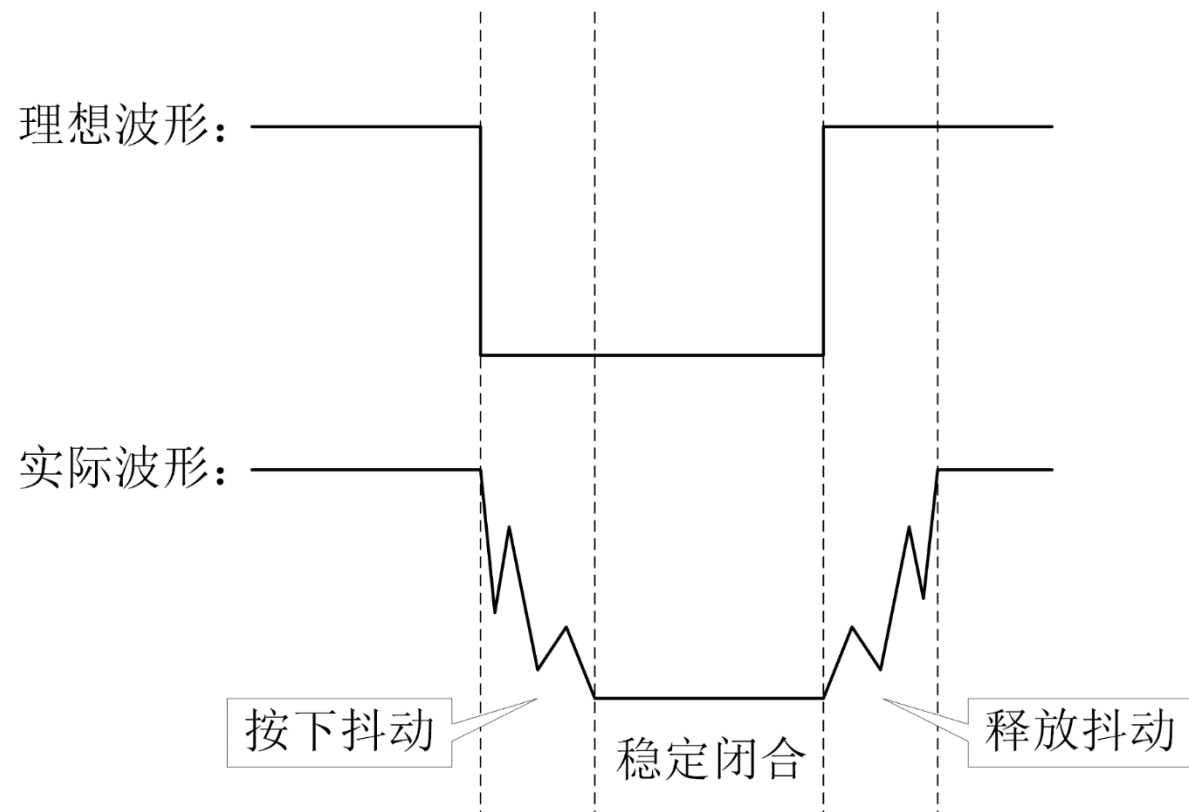


Circuit Diagram

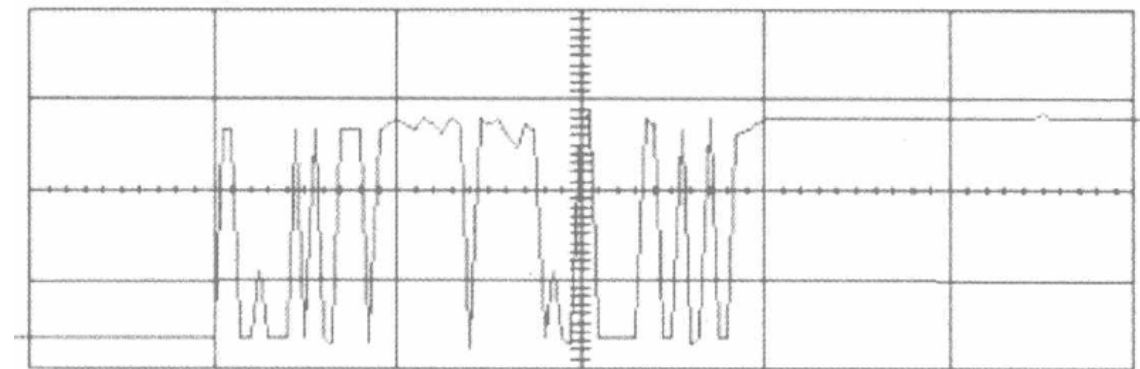


6.3 思考与改进

按键抖动的现象



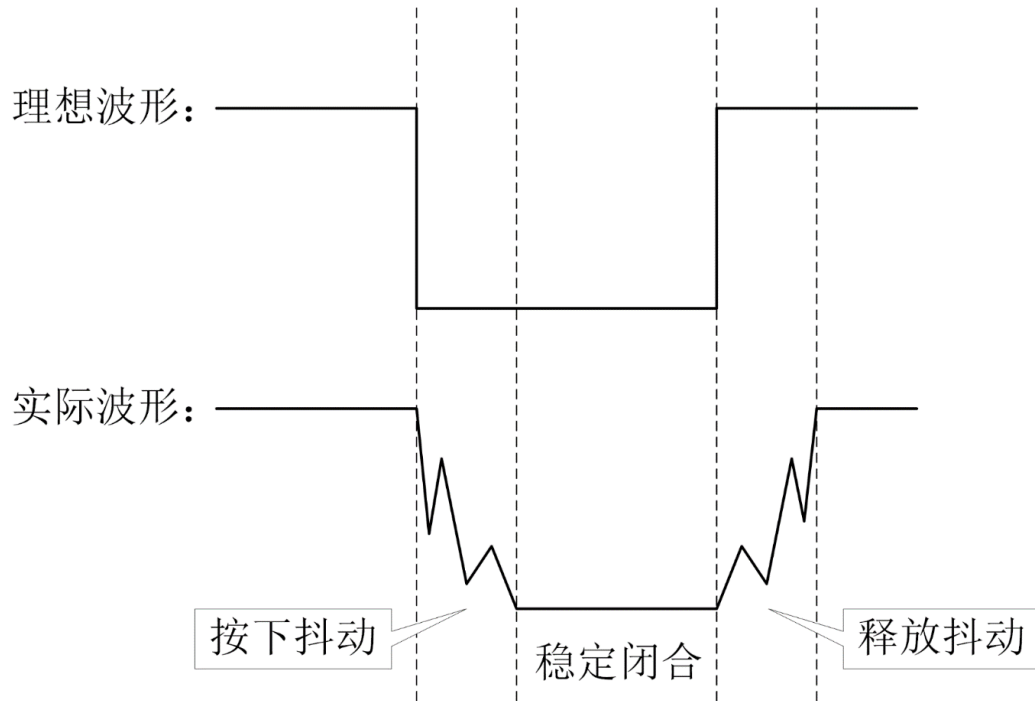
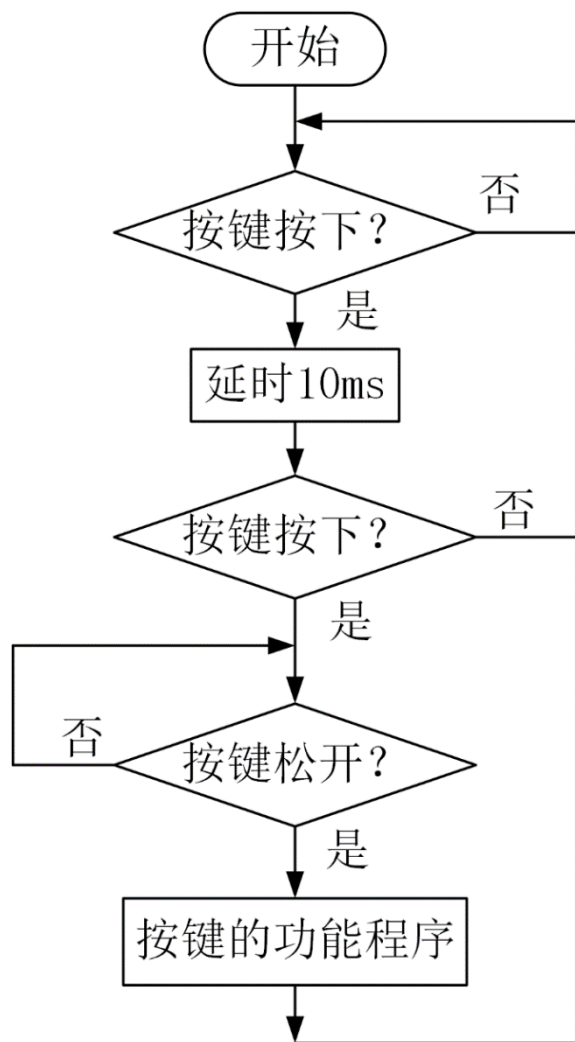
小按钮开关闭合瞬间



小型继电器闭合瞬间

6.3 思考与改进

软件消抖

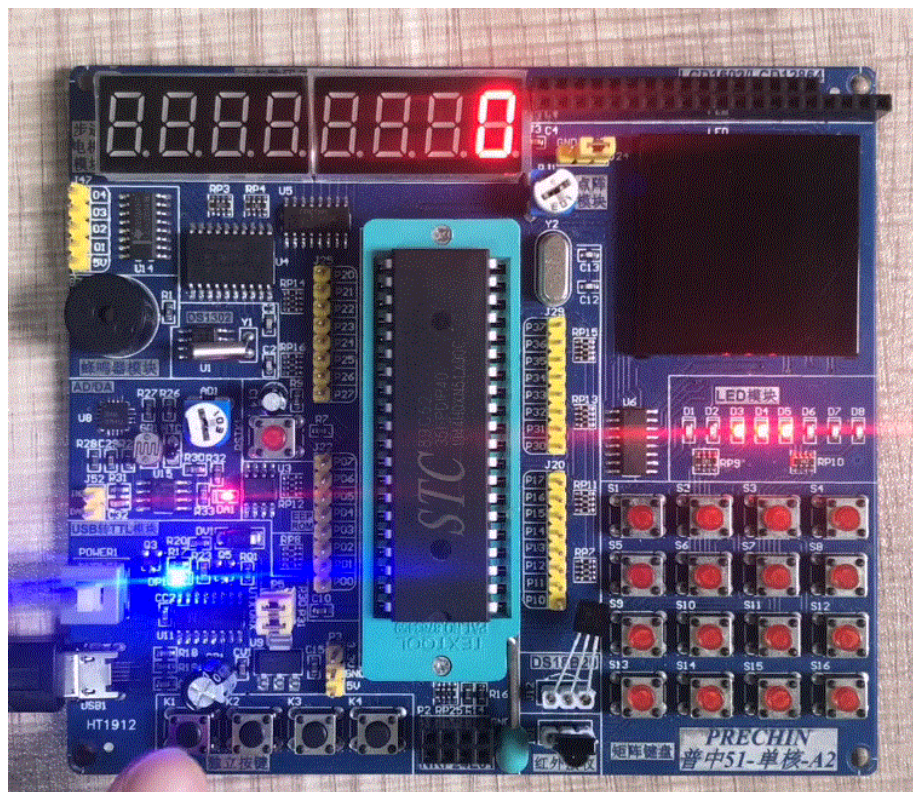


```
if (key==0)
{
    delay(1100);
    if (key==0)
    {
        //写按下按键要做的代码
    }
    while (!key);
    //写松开按键要做的代码
}
```

程序设计

6.3 思考与改进

软件消抖后的程序



效果展示

```
#include <regx51.h>
#define uint unsigned int
#define uchar unsigned char
sbit key=P3^1;
uchar code table[]={0x3f,0x06,0x5b,
0x4f,0x66,0x6d,0x7d,0x07,0x7f,0x6f,
0x77,0x7c,0x39,0x5e,0x79,0x71};
void delay(uint i);
void main()
{
    uchar num=0;
    P2=0xe3;
    P0=table[0];
    key=1;                                     //读引脚前置高电平
    while(1)
    {
        if(key==0)                           //检测按键是否按下
        {
            delay(1100);                      //延时消抖
            if(key==0)                        //再次检测按键是否按下
            {
                num++;
                if(num>9)
                    num=0;
                while(key==0);                //松开按键才加1
                P0=table[num];
            }
        }
    }
}
void delay(uint i)
{
    while(i--);
}
```


6.4 实战作业

1. 独立按键控制数码管**减一**显示（9~0）
2. 独立按键控制一个LED灯亮灭