

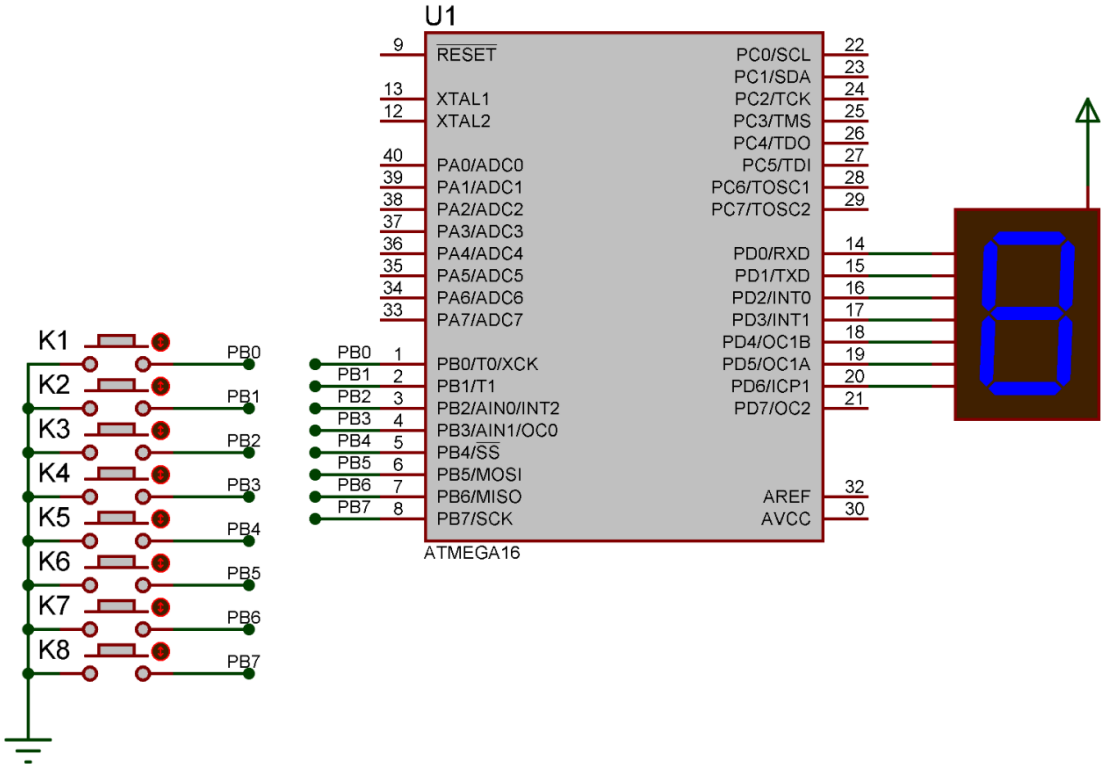
# 实验 6：端口按键判断（按键显示数字）

## 1. 试验描述

将 8 个按键从 1~8 进行编号，如果其中一个键按下，则在 LED 数码管上显示相应的键值。

## 2. 系统框图

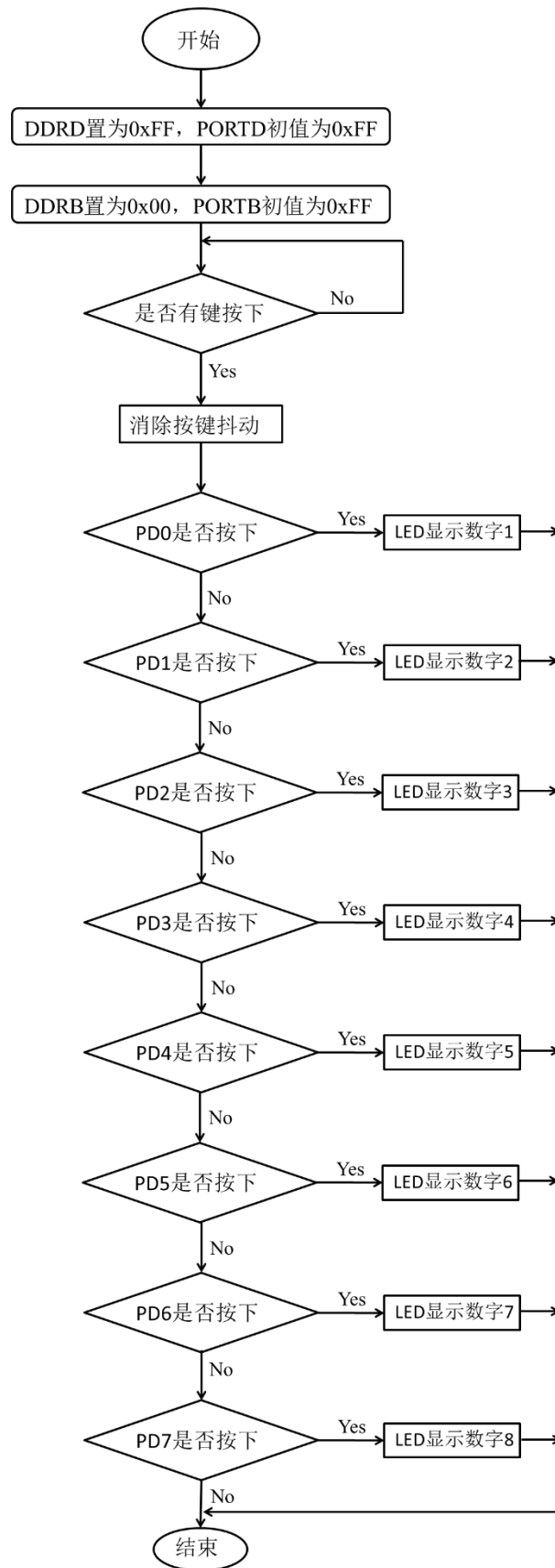
### ➤ 硬件电路



### ➤ 元件清单

单片机 ATmega16	按键 BUTTON
共阳极 LED 数码管 7SEG-COM-AN-BLUE	

➤ 软件流程



## ➤ 键盘工作原理

键盘是由按键构成的，键的闭合与否通常用高、低电平来进行检测。键闭合时，该键为低电平；键断开时，该键为高电平。

键的闭合与断开是利用其机械弹性来实现的。由于机械弹性的作用，键在闭合与断开的瞬间均有抖动过程，抖动的时间一般为 5~10ms。按键的稳定闭合期由操作人员按键动作所决定，为了使 CPU 对按键的一次闭合仅做一次按键输入处理，必须去抖动。去抖动有硬件方法和软件方法两种。例如，采用滤波电路防抖、RS 触发器构成的双稳态去抖电路，这些方法属于硬件去抖法。软件去抖法就是指检测到有键按下时，先执行一个 10~20ms 的延时子程序，再确认该键是否仍保持闭合状态，若仍闭合则确认为此键按下，由此便消除了抖。

在键盘操作过程中，当有两个或两个以上的键被同时按下或先后按下时，到底哪个按键有效完全取决于系统开发者的设计。

独立式键盘直接由 I/O 口线构成。每个按键接一根输入线，占用一根 I/O 口线，各键的工作状态互不影响，如图 1 所示。

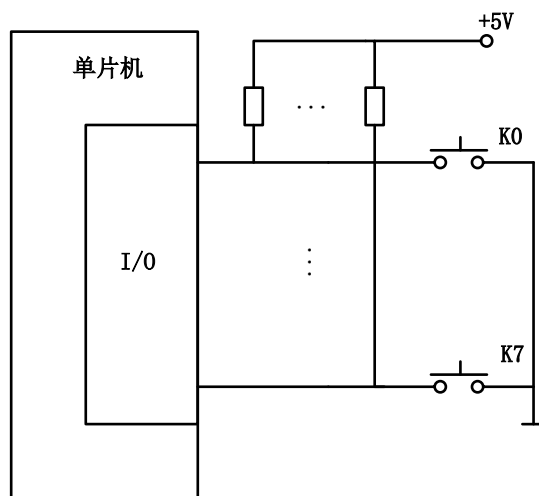


图 1 独立式键盘电路

### 3. 程序代码

#### ➤ ICAVR 程序

```
#include <iom16v.h>

#define uchar unsigned char
#define uint unsigned int

//共阳极 LED0~F 的段码
const uchar tab[16]={0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,
                    0x80,0x90,0x88,0x83,0xC6,0xA1,0x86,0x8E};

void delay(uint n)
{
    uint i;
    for(i=0;i<n;i++);
}

void main(void)
{
    uchar key;
    DDRD=0xFF;
    PORTD=0xFF;
    DDRB=0x00;
    PORTB=0xFF;
    while(1)
    {
        while(PINB==0xFF);           //等待键按下
        {
            delay(2500);              // 延时出去抖动
            while(PINB==0xFF);
            {
                key=PINB;              //读取键值
                switch(key)
                {
                    case 0xfe :    PORTD=tab[1]; break;
                    case 0xfd :    PORTD=tab[2]; break;
                    case 0xfb :    PORTD=tab[3]; break;
                    case 0xf7 :    PORTD=tab[4]; break;
                    case 0xef :    PORTD=tab[5]; break;
                    case 0xdf :    PORTD=tab[6]; break;
                    case 0xbf :    PORTD=tab[7]; break;
                }
            }
        }
    }
}
```

```
case 0x7f : PORTD=tab[8]; break;
```

## CVAVR 程序

```
#include<mega16.h>

#define uchar unsigned char
#define uint unsigned int

//共阳极 LED0~F 的段码
flash uchar tab[16]={0xC0,0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,
                     0x80,0x90,0x88,0x83,0xC6,0xA1,0x86,0x8E};

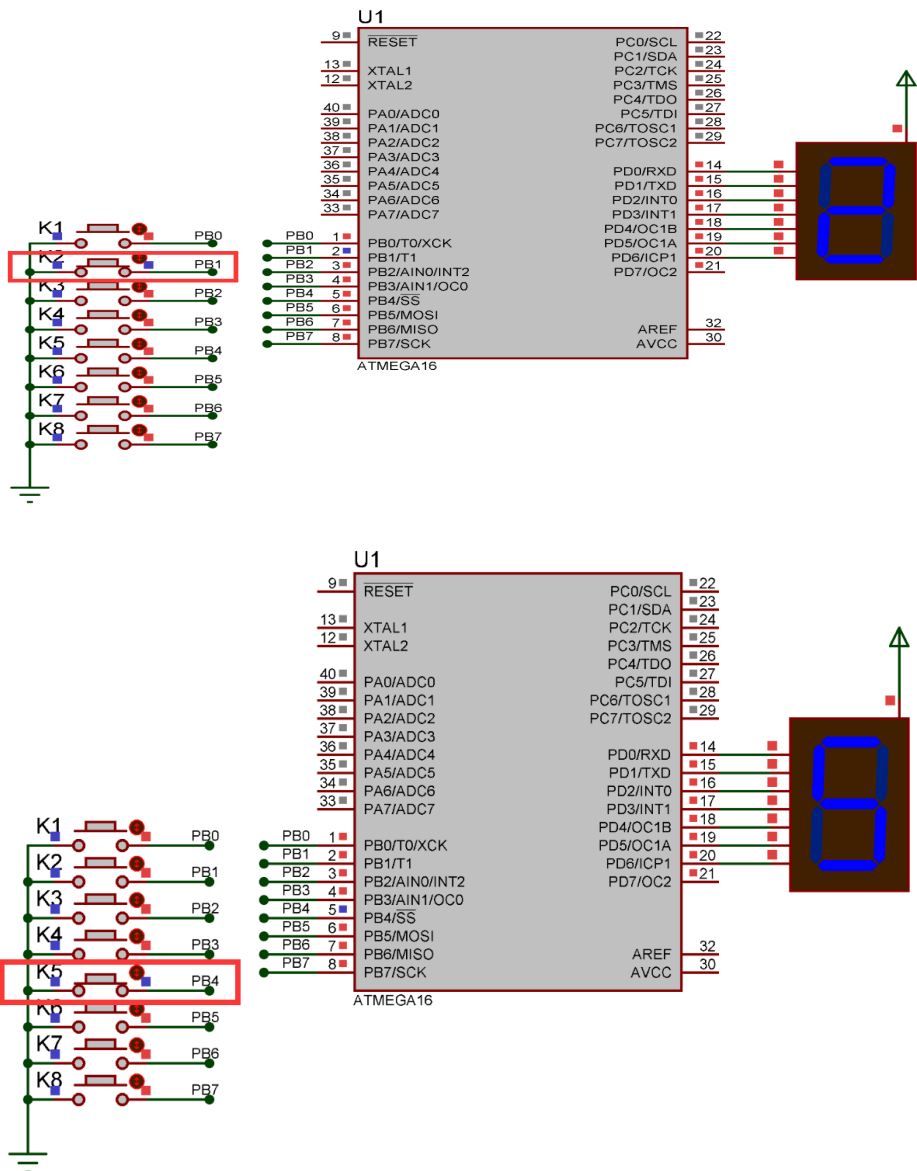
void delay(uint n)
{
    uint i;
    for(i=0;i<n;i++);
}

void main(void)
{
    uchar key;
    DDRD=0xFF;
    PORTD=0xFF;
    DDRB=0x00;
    PORTB=0xFF;
    while(1)
    {
        while(PINB==0xFF);           //等待键按下
        {
            delay(2500);              // 延时出去抖动
            while(PINB==0xFF);
            {
                key=PINB;              //读取键值
                switch(key)
                {
                    case 0xfe :    PORTD=tab[1]; break;
                    case 0xfd :    PORTD=tab[2]; break;
                    case 0xfb :    PORTD=tab[3]; break;
                    case 0xf7 :    PORTD=tab[4]; break;
                }
            }
        }
    }
}
```

```
case 0xef : PORTD=tab[5]; break;
case 0xdf : PORTD=tab[6]; break;
case 0xbf : PORTD=tab[7]; break;
case 0x7f : PORTD=tab[8]; break;
}
}
}
}
}
```

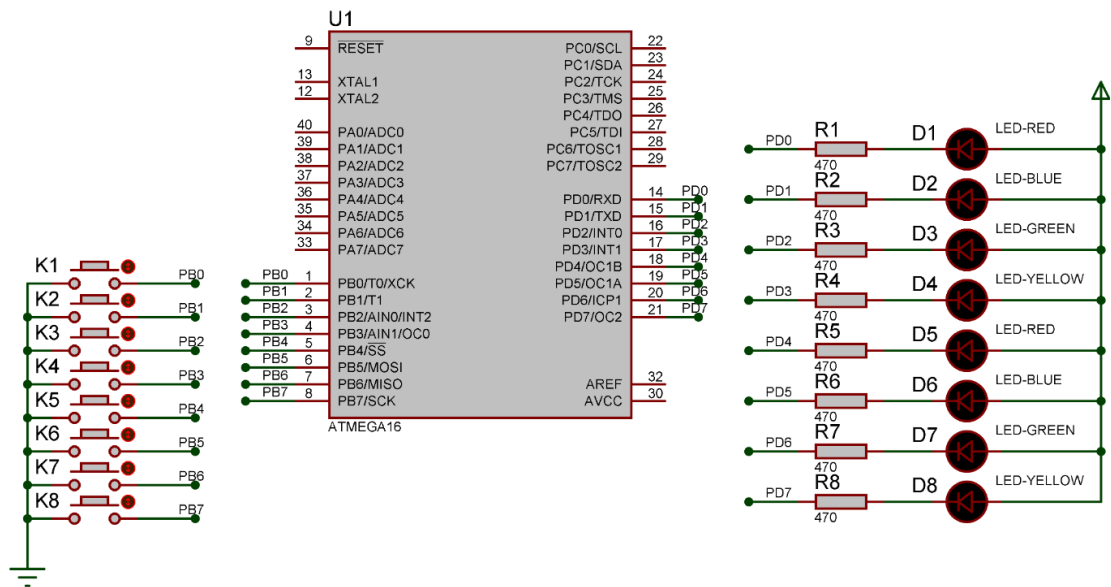
## 4. 仿真结果

开始运行时，LED 无显示，当按下某键时，将显示相应的数值。如，按下 5 键，运行仿真效果图如下。



练习一:将数码管换成 LED 小灯，用 8 个按键控制 8 个 LED 小灯的亮灭。

## ➤ 硬件电路



## ➤ ICAVR 程序

```
#include<iom16v.h>

#define uchar unsigned char
#define uint unsigned int

void s_ms(unsigned int t)
{
    for (;t>1;t--)
    {

    }
}

void key()
{
    if((PINB&(1<<PB0))==0)
    {
        PORTD=~(1<<PD0);
    }
    if((PINB&(1<<PB1))==0)
    {
        PORTD=~(1<<PD1);
    }
}
```

```

        if((PINB&(1<<PB2))==0)
        {
            PORTD=~(1<<PD2);
        }
        if((PINB&(1<<PB3))==0)
        {
            PORTD=~(1<<PD3);
        }
        if((PINB&(1<<PB4))==0)
        {
            PORTD=~(1<<PD4);
        }
        if((PINB&(1<<PB5))==0)
        {
            PORTD=~(1<<PD5);
        }
        if((PINB&(1<<PB6))==0)
        {
            PORTD=~(1<<PD6);
        }
        if((PINB&(1<<PB7))==0)
        {
            PORTD=~(1<<PD7);
        }
    }
}

void main()
{
    DDRD=0xFF;
    PORTD=0xFF;
    DDRB=0x00;
    PORTB=0xFF;
    while(1)
    {
        key();
    }
}

```

## ➤ CAVR 程序

```

#include<iom16v.h>

#define uchar unsigned char
#define uint unsigned int

```



```
void s_ms(unsigned int t)
{
    for (;t>1;t--)
    {

    }
}

void key()
{
    if((PINB&(1<<PB0))==0)
    {
        PORTD=~(1<<PD0);
    }
    if((PINB&(1<<PB1))==0)
    {
        PORTD=~(1<<PD1);
    }
    if((PINB&(1<<PB2))==0)
    {
        PORTD=~(1<<PD2);
    }
    if((PINB&(1<<PB3))==0)
    {
        PORTD=~(1<<PD3);
    }
    if((PINB&(1<<PB4))==0)
    {
        PORTD=~(1<<PD4);
    }
    if((PINB&(1<<PB5))==0)
    {
        PORTD=~(1<<PD5);
    }
    if((PINB&(1<<PB6))==0)
    {
        PORTD=~(1<<PD6);
    }
    if((PINB&(1<<PB7))==0)
    {
        PORTD=~(1<<PD7);
    }
}

void main()
```

```

{
    DDRD=0xFF;
    PORTD=0xFF;
    DDRB=0x00;
    PORTB=0xFF;
    while(1)
    {
        key();
    }
}

```

## ➤ 仿真结果

