

8) 某放大电路中使用的三极管的极限参数为 $P_{CM} = 100\text{mW}$, $I_{CM} = 20\text{mA}$, $U_{(BR)CEO} = 15\text{V}$ 。

以下哪些情况下, 三极管不能正常工作 ()。

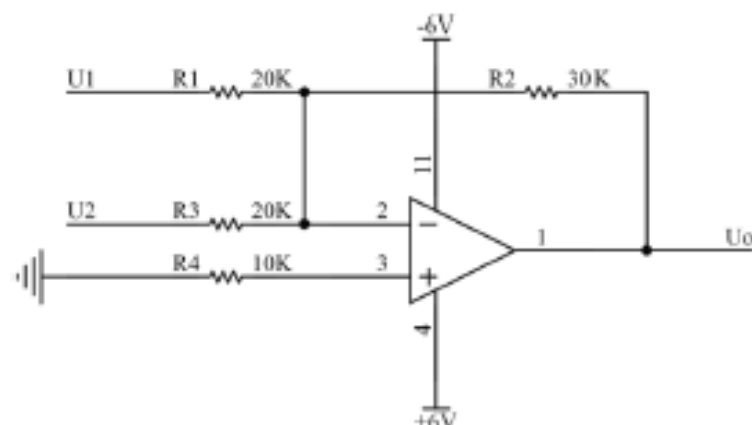
A. $U_{CE} = 3\text{V}$, $I_C = 15\text{mA}$

B. $U_{CE} = 2\text{V}$, $I_C = 40\text{mA}$

C. $U_{CE} = 6\text{V}$, $I_C = 20\text{mA}$

D. $U_{CE} = 9\text{V}$, $I_C = 10\text{mA}$

9) 电路如下图所示, 输入电压 $U_1=0.4\text{V}$, $U_2=0.8\text{V}$, 计算输出电压 U_o 的值为 ()。



A. 1.2V

B. -1.2V

C. -6V

D. -1.8V

10) 下列说法中正确的是 ()。

A. IAP15F2K61S2 单片机可以通过串口实现在线仿真功能。

B. 单片机竞赛板在 IO 和 MM 模式下, 均可实现对数码管和 LED 指示灯的分别操作, 互不影响。

C. 对 DS1302 进行单字节写操作时, 数据在时钟线 SCLK 下降沿写入 DS1302。

D. I2C 总线的启动信号和停止信号, 只能由主器件发起。[csdn.net/Xiaomo_haa](https://www.csdn.net/Xiaomo_haa)

第十届 蓝桥杯 单片机设计与开发项目 省赛

第二部分 程序设计试题 (70 分)

1、基本要求

- 1.1 使用大赛组委会提供的国信长天单片机竞赛实训平台，完成本试题的程序设计与调试。
- 1.2 选手在程序设计与调试过程中，可参考组委会提供的“资源数据包”。
- 1.3 **请注意：**程序编写、调试完成后选手应通过考试系统提交完整、可编译的 Keil 工程文件。选手提交的工程文件应是最终版本，要求 Keil 工程文件以准考证号（8 位数字）命名，工程文件夹内应包含以准考证号命名的 hex 文件，该 hex 文件是成绩评审的依据。不符合以上文件提交要求的作品将被评为零分或者被酌情扣分。
- 1.4 请勿上传与作品工程文件无关的其它文件。

2、竞赛板配置要求

- 2.1 将 IAP15F2K61S2 单片机内部振荡器频率设定为 12MHz。
- 2.2 键盘工作模式跳线 J5 配置为 BTN 独立按键模式。
- 2.3 扩展方式跳线 J13 配置为 IO 模式。
- 2.4 **请注意：**选手需严格按照以上要求配置竞赛板，编写和调试程序，不符合以上配置要求的作品将被评为零分或者被酌情扣分。

3、硬件框图

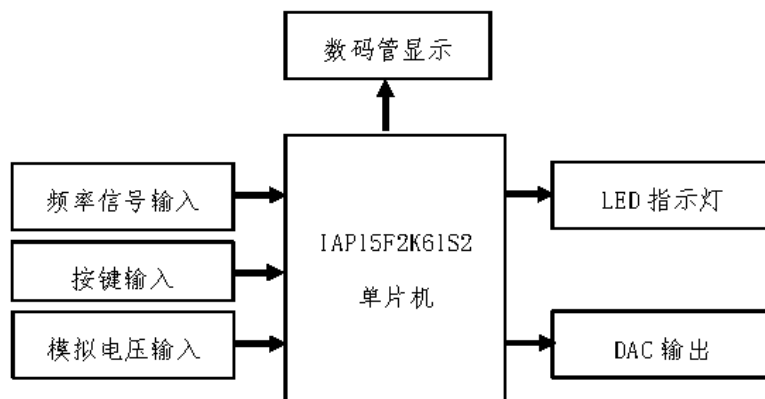


图 1 系统硬件框图

4、功能描述

4.1基本功能

- 1) 测量竞赛板上电位器 RB2 输出的模拟电压信号和 NE555 模块输出的频率信号，以数码管、LED 等外围设备进行数据呈现。
- 2) 频率测量功能需将竞赛板 J3-SIGNAL 引脚与 P34 引脚短接。(P34 与 SIGNAL 的短接可以使用竞赛板上超声/红外切换等与本试题功能要求无关的跳线帽完成)。
- 3) 使用 PCF8591 测量电位器 RB2 的输出电压，并根据试题要求通过其 DAC 功能输出该电压值。
- 4) 电压、频率数据刷新时间要求
 - 电压数据刷新时间 ≤ 0.5 秒。
 - 频率数据刷新时间 ≤ 1 秒。
- 5) 电压、频率数据测量范围要求
 - 电压数据测量范围：电位器 RB2 输出的最小电压值到最大电压值。
 - 频率数据测量范围：NE555 模块输出的最低频率到最高频率值。

4.2显示功能

1) 频率显示界面

频率显示界面如图 2 所示，显示内容包括提示符 F 和频率值，频率数据单位为 Hz。

| | | | | | | | |
|-----|----|-----------|---|---|---|---|---|
| F | 8 | 8 | 8 | 6 | 0 | 0 | 0 |
| 提示符 | 熄灭 | 频率：6000Hz | | | | | |

图 2 频率测量显示界面

备注：如上图所示，频率数据显示使用 6 位数码管，当显示的数据长度不足 6 位时，未使用到的数码管位应熄灭。

2) 电压显示界面

电压显示界面如图 3 所示，显示内容包括提示符 U 和电位器 RB2 输出的电压值，电压测量结果保留小数点后两位有效数字。

| | | | | | | | |
|---|---|---|---|---|----|---|---|
| U | 8 | 8 | 8 | 8 | 3. | 4 | 1 |
|---|---|---|---|---|----|---|---|

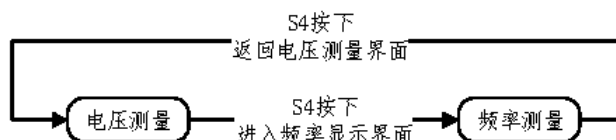
| | | |
|-----|--------|-----------|
| 提示符 | 未启用：熄灭 | 电压值：3.41V |
|-----|--------|-----------|

图3 电压测量显示界面

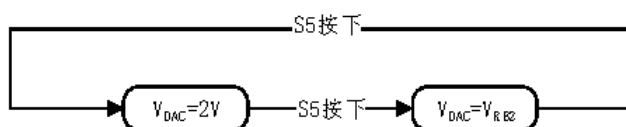
4.3 按键功能

1) 按键功能说明

- S4：定义为“显示界面切换”按键，按下 S4 按键，切换选择频率显示界面和电压显示界面，按键 S4 切换模式如下图所示：



- S5：定义为 PCF8591 DAC “输出模式切换”按键，按下 S5，DAC 输出电压跟随电位器 RB2 输出电压 V_{RB2} 变化而变化，保持与 V_{RB2} 电压值一致；再次按下 S5，DAC 输出固定电压 2.0V，不再跟随电位器 RB2 输出电压变化。按键 S5 工作模式如下图所示：

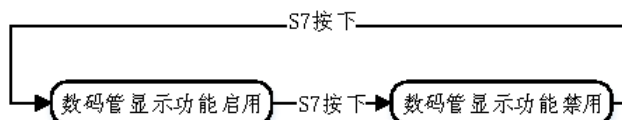


- S6：定义为“LED 指示灯功能控制”按键，按下 S6 按键，关闭或打开 LED 指示灯指示功能。按键 S6 工作模式如下图所示：



备注：关闭 LED 指示灯功能状态下，所有 LED 指示灯熄灭。

- S7：定义为“数码管显示功能控制”按键，按下 S7 按键，关闭或打开数码管显示功能。按键 S7 工作模式如下图所示：



备注：关闭数码管状态下，所有数码管熄灭。

4.4 LED 指示灯功能

- 1) 电压测量功能指示：L1 点亮，L2 熄灭
- 2) 频率测量功能指示：L1 熄灭，L2 点亮

3) 指示灯 L3 功能:

| 电位器 RB2 输出电压 (V_{RB2}) | L3 指示灯状态 |
|----------------------------|----------|
| $V_{RB2} < 1.5V$ | 熄灭 |
| $1.5V \leq V_{RB2} < 2.5V$ | 点亮 |
| $2.5V \leq V_{RB2} < 3.5V$ | 熄灭 |
| $V_{RB2} \geq 3.5V$ | 点亮 |

4) 指示灯 L4 功能:

| 信号频率 (F_{out}) | L4 指示灯状态 |
|-----------------------------|----------|
| $F_{out} < 1KHz$ | 熄灭 |
| $1KHz \leq F_{out} < 5KHz$ | 点亮 |
| $5KHz \leq F_{out} < 10KHz$ | 熄灭 |
| $F_{out} \geq 10KHz$ | 点亮 |

5) 指示灯 L5 功能: DAC 输出固定电压 (2.0V) 时, L5 熄灭, DAC 输出电压跟随 RB2 电位器输出电压变化时, L5 点亮。

6) 本试题未涉及的 LED 指示灯应处于熄灭状态。

4.5 初始状态说明

- 1) 初始状态上电默认处于电压测量状态, 数码管显示和 LED 指示功能启用。
- 2) 初始状态上电默认 PCF8591 DAC 芯片输出固定电压值 2.0V。

参考代码:

```
#include "STC15F2K60S2.H"
#include "MAIN.H"
#include "HAREWARE.H"
#include "DIG.H"
#include "I2C.H"
#include "DELAY.H"
#include "LED.H"
```

```
sbit s4 = P3^3;//按键 S4
sbit s5 = P3^2;//按键 S5
sbit s6 = P3^1;//按键 S6
sbit s7 = P3^0;//按键 S7
```

```
unsigned int PCF_NUM = 0;//电压值
unsigned char led_mode = 0xff;          //LED 灯点亮模式
unsigned int f_num = 0;//频率数值
unsigned char j = 0;//频率测量里面的等待中断标志
unsigned char diya_num = 0;//储存模数转换返回的电压值
unsigned char timer = 0;
```

```
bit s4_mode = 1;//1 为电压测量模式，0 为频率测量模式
bit s5_mode = 0;//1 为输出 2.0v 电压，为 0 跟随 PCF8591 三通道电压输出
bit s6_mode = 1;//1 为开启 LED 灯显示，0 为关显示
bit s7_mode = 1;//1 为开启数码管显示，0 为关显示
bit gensui = 0;//0 为不跟随电压输出，1 为跟随电压输出
```

```
unsigned char useg_num[]={0x3e,0,0,0,0,0,0,0};//电压数值数组
unsigned char fseg_num[]={0x71,0,0,0,0,0,0,0};//频率数值数组
```

```
void key_scan();//按键扫描函数原型
unsigned char dian_ya_ce_liang();//电压测量函数原型
void TimerInit();//定时计数初始化函数原型
```

```
/*
*****
*****函数*****
*****
*/
```

```
void main();//主函数
{
    HAREWARE_init();//关闭外部设备
    DIG_init();//数码管初始化
    LED_init();//LED 灯初始化
    TimerInit();//定时器 1 初始化
    P20 = 1;
    while(1)//等待中断
    {
```

```

key_scan();//按键扫描

if(~s5_mode)//判断是否是电压跟随模式
{
    PCF_DA(103); //没有跟随
    gensui = 0;  //没有跟随
}
else
{
    gensui = 1;          //跟随电压
}

if(s4_mode)//判断是否为电压测量模式
{
    diya_num = dian_ya_ce_liang();
    led_mode = 0xfe;//电压测量模式指示灯亮
    if(((diya_num * 0.01953125 >= 1.5) && (diya_num * 0.01953125 < 2.5)) || (diya_num *
0.01953125 >= 3.5))
    {
        led_mode &= 0xfb;//满足条件第三个指示灯亮
    }
    if(gensui)
    {
        led_mode &= 0xef;//电压跟随第五个指示灯亮
    }
    PCF_NUM = diya_num * 1.953125;//变成十进制的数字,把小数乘一百
    useg_num[5]=DIG_CODE[PCF_NUM/100];//百位数字
    PCF_NUM = PCF_NUM % 100;
    useg_num[6]=DIG_CODE[PCF_NUM/10];//十位数字
    PCF_NUM %=10;
    useg_num[7]=DIG_CODE[PCF_NUM];//个位数字
    LED_show(led_mode);//LED 等显示
    DIG_show(useg_num);//数码管显示
    if(gensui)//判断是否是电压跟随模式
    {
        PCF_DA(diya_num); //没有跟随
    }
}
else//频率测量模式
{
    TR1 = 1;//开始计时
    TR0 = 1;//开始计数
    j = 1;//等待中断标志置 1
    while(j);//等待中断
}
}
}

```

////////////////////////////////////


```
////////////////////////////////////
```

```
void TimerInit()//定时计数器初始化函数
```

```
{
    AUXR &= 0x3F;//12 分频
    TMOD &= 0x00;//TMOD（定时/计数器模式寄存器）清零
    TMOD |= 0x15;//T0 设置为模式 1，计数；T1 设置为模式 1，定时
    TH0 = TL0 = 0;//计数器清零
    TL1 = 0xF0;           //设置定时初值
    TH1 = 0xD8;           //设置定时初值
    EA = 1;//开总中断
    ET1 = 1;//开定时器 1 中断
}
```

```
////////////////////////////////////
////////////////////////////////////
```

```
unsigned char dian_ya_ce_liang()
```

```
{
    return PCF_AD(3);//返回模数转换值
}
```

```
////////////////////////////////////
////////////////////////////////////
```

```
void key_scan()//键盘扫描函数
```

```
{
    P3 |= 0x0f;//P3 低 4 位设为全 1
    if(~s4)//判断 S4 按键
    {
        delay1ms(5);//按键消抖
        if(~s4)//判断 S4 按键
        {
            s4_mode = ~s4_mode;
        }
    }

    if(~s5)//判断 S5 按键
    {
        delay1ms(5);//按键消抖
        if(~s5)//判断 S5 按键
        {
            if(s4_mode)//如果在电压测量模式则改变电压跟随标志
            {
                s5_mode = ~s5_mode;//s5_mode 取反
            }
        }
    }
}
```

```

if(~s6)//判断 S6 按键
{
    delay1ms(5);//按键消抖
    if(~s6)//判断 S6 按键
    {
        s6_mode = ~s6_mode;
        if(~s6_mode)
        {
            LED_init();//关 LED 灯
        }
    }
}

if(~s7)//判断 S7 按键
{
    delay1ms(5);//按键消抖
    if(~s7)//判断 S7 按键
    {
        s7_mode = ~s7_mode;//取反
        if(~s7_mode)//如果数码管开关标志为 0，关数码管显示
        {
            DIG_init();//关数码管
        }
    }
}

while(~s4||~s5||~s6||~s7)//松手检测
{
    if(s4_mode)//如果为电压测量模式，数码管显示电压值，防止出现数码管黑屏现象
    {
        DIG_show(useg_num);//数码管显示电压测量值
        LED_show(led_mode);//LED 显示电压测量模式
    }
    else//如果为频率测量模式，数码管显示频率值，防止出现数码管黑屏现象
    {
        DIG_show(fseg_num);//数码管显示频率测量值
        LED_show(led_mode);//LED 显示频率测量模式
    }
}

}
/*//////////////////////////////////////
//////////定时器 1 中断函数//////////
//////////////////////////////////////*/
void T1()interrupt 3//定时器 1 中断函数
{
    unsigned char i = 0;
    unsigned int f_num2 = 0;

```

```

P20 = ~P20;
TR0 = 0;//停止计数
TR1      = 0;//停止定时
TL1 = 0xF0;//设置定时初值
TH1 = 0xD8;//设置定时初值
timer++;//计数器加 1
j = 0;//等待中断标志清零
if(timer == 20)
{
    f_num = (TH0 * 256 + TL0) * 5; //计算频率
    timer = 0;
    TL0 = TH0 = 0;//计数器清零
    for(i=7;i>2;i--)//重新初始化频率数值数组
    {
        fseg_num[i] = 0;//除数组的第一位和第二位其他的初始化为 0
    }
    if((f_num >= 1000 && f_num < 5000) || (f_num >= 10000))
    {
        led_mode = 0XF5;//满足条件 led4 亮
    }
    else
    {
        led_mode = 0XFD;
    }
    for(i = 7;i>2 && f_num != 0;i--)
    {
        fseg_num[i] = DIG_CODE[f_num % 10];
        f_num /= 10;
    }
}
LED_show(led_mode);
DIG_show(fseg_num);
}

```