

考试形式: 机试/闭卷 学习通考试 考试时间: 90 分钟 考试题型如下

- 一、单选题 (每题 2 分, 共 20 分) 注: 10 道选择题
- 二、填空题 (每题 2 分, 共 20 分) 注: 20 道填空题
- 三、简单题 (每题 6 分, 共 30 分) 注: 5 道简答题
- 四、编程题 (每题 10 分, 共 30 分) 注: 3 道编程题

考试复习内容:

一、选择题

1.语句 P1DIR |= 0x03 的作用是 (B)

- A、将 P1_0 和 P0_1 作为普通 IO 口
- B、将 P1_0, P1_1 设置为输出
- C、将 P1_0, P1_1 设置为输入
- D、将 P1_0, P1_1 设置为连接外设

2.下列 (D) 不属于 ZigBee 的拓扑结构

- A、星型 B、树形 C、网状 D、总线型

3.使用 10 位的 ADC 可以有 (B) 个量化值

- A、512 B、1024 C、2048 D、4096

4.CC2530 串行通信接口 USART1 的控制和状态控制寄存器是 (B)

- A、U1BAUD B、U1CSR C、U1BUF D、U1GCR

5.(B)不具有蓝牙通信功能

- A. ESP32 C2 B.ESP32 S2 C.ESP32 C3 D.ESP32 S3

6.ESP32 是由 (C) 开发的主流的嵌入式 WiFi 蓝牙 SoC (System-on-Chip)。

- A.德州仪器 B.MicroSoft C.乐鑫科技 D.英伟达

二、填空题

1. 将 ESP32 的 GPIO25 设置为输出模式, 并输出低电平的代码分别为
(`pinMode(25,OUTPUT);`)、(`digitalWrite(25,LOW);`) 注意! 大小写不能出错

2. 如果将 CC2530 的 P1 设置为通用 IO 口, 需要设置 (P1SEL) 器

3. 1 个 11 位的 ADC,若参考电压为 5V, 其分辨率为(2.44) mv(保留两位小数)

计算方法
$$\text{分辨率} = \frac{\text{电压 (毫伏)}}{2^{\text{ADC 的位数}}} \quad \text{例: } 5000 / 2^{11} = 2.44$$

4. ESP32 的 ADC1 支持 (8) 个通道, ADC2 支持 (10) 个通道

5. CC2530 微控制器的内核是增强型 8051, 一个有 (21) 个数字 IO 引脚, 其中 P0 端口有 (8) 个引脚, P1 端口有 (8) 个引脚, P2 端口有 (5) 个引脚

6.CC2530 外部中断函数服务格式为 (`#pragma vector =中断向量, 紧接着是中断服务函数`)

7.串行通信每 0.5 秒钟传输 5760 个字符,每个字符有 10 个代码位,其通信波特率为(

$$115200 \text{ bit/s}$$
) 注意! 一定要带单位 bit/s

计算方法
$$\text{波特率} = \frac{1}{\text{时间}} \times \text{字符数} \times \text{代码位} \quad \text{注意: 时间单位为秒}$$

三、简答题

1. 1.ESP32 的功能主要有哪些? (至少写 6 点)

1. Wifi
2. 蓝牙
3. CPU 和储存
4. 时钟和定时器
5. 高级外设接口
6. 安全机制

2. 简述 CC2530 中断处理过程。

1. 中断检测
2. 中断优先级判断
3. 中断响应
4. 中断服务程序执行
5. 清除中断标志
6. 恢复执行

3.简述 ZigBee 网络中协调器、路由器、终端设备的作用。

协调器:

- 1.启动与配置网络;
- 2.维护网络;
- 3.与外部网络通信;
- 4.管理节点。

路由器:

- 1.允许其它的节点加入网络;
- 2.转发数据包;
- 3.辅助它的孩子节点进行网络通信。

终端设备:

- 1.发送和接收数据;
- 2.控制其他终端设备

4 写出 ESP32 使用蓝牙进行通信的全过程

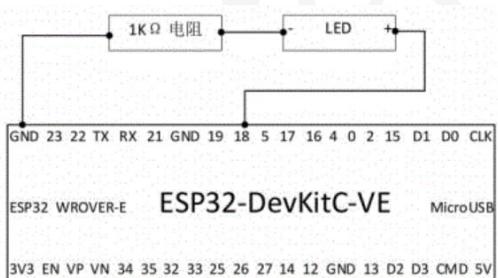
1. 初始化蓝牙
2. 配置蓝牙角色
3. 发现和连接设备
4. 建立通信
5. 数据传输
6. 数据处理
7. 断开连接 (视使用场景而定)

5.ESP32 的串口使用可以分哪几部分?

1. 串口初始化
2. 发送数据
3. 接收数据
4. 串口事件处理
5. 串口调试

四、编程题

1. 电路如图所示, 请使用 Arduino 或 VS Code 开发环境实现, LED 亮 1 秒, 灭 1 秒的程序功能



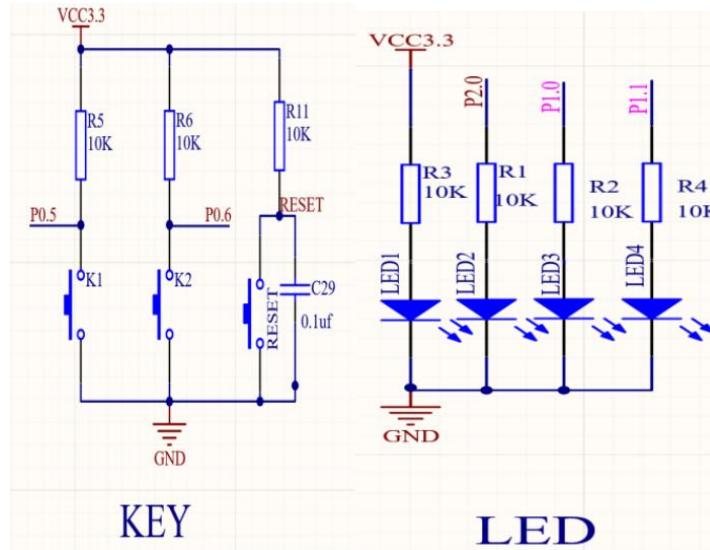
```
#define LED 18
```

```
void setup()
{
    PinMode(LED,OUTPUT);
}
```

```
void loop()
```

```
{  
    digitalWrite(LED,HIGH);  
    delay(1000);  
    digitalWrite(LED,LOW);  
    delay(1000);  
}
```

2. 如图所示，编写程序实现如下功能：第1次按下按键K2电亮1盏LED灯，第2次按下K2点亮两盏灯，第3次按下K2点亮3盏灯，第4次按下K2熄灭三盏灯，第5次按下K2重复第1次的动作，以此反复



```
#include <iocc2530.h>

#define uint unsigned int
#define uchar unsigned char

#define LED2 P2_0
#define LED3 P1_0
#define LED4 P1_1

#define KEY2 P0_6

void delay(uint temp)
{
    for(int j = 100; j > 0; j--)
        for(int i = temp; i > 0;
    }

void LedInit()
{
```

```
P1DIR |= 0X03;
P2DIR |= 0X01;
LED2 = 0;
LED3 = 0;
LED4 = 0;
}

void KeyInit()
{
    P0DIR &= 0X9F;
    P2INP &= 0XDF;
    P0INP &= 0X9F;
}

uchar Button()
{
    if(KEY2 == 0)
    {
        delay(5);
        if(KEY2 == 0)
            return 1;
        else
            return 0;
    }
}

void LedControl(uchar temp)
{
    uchar select = temp;
    if(AddNumber == 5)
        AddNumber = 1;
    switch(select)
    {
        case 1:{LED2 = 1; LED3 = 0; LED4 = 0; break;}
        case 2:{LED2 = 0; LED3 = 1; LED4 = 0; break;}
        case 3:{LED2 = 0; LED3 = 0; LED4 = 1; break;}
        case 4:{LED2 = 0; LED3 = 0; LED4 = 0; break;}
        default: {break;}
    }
}

void main()
{
    LedInit();
```

```

KeyInit();

uchar AddNumber = 0; //用于记录按钮按下次数

while(1)
{
    if(Button() == 1)
    {
        AddNumber++;
        LedControl(AddNumber);
        delay(2000);
    }
}
}

```

3. 请写出 ESP32 连接 Wifi 的代码。要求:

(1) 正在连接的时候 LED 灯闪烁 (假设 LED 连接的是 GPIO12)、并在串口输出“WIFI 正在连接”

(2) 连接上 WIFI 之后, 在串口输出“WiFi 已连接”并且显示 IP 地址、LED 灯常亮

注意: 代码中如果出现了 WiFi 账号和密码, 请用自己的姓名拼音表示 WiFi 账号, 自己的学号表示 WiFi 密码

```

#include <WiFi.h>

#define LED 12

const char *ssid = "xiaoyifan"
const char *password = "202102165545"

void setup()
{
    Serial.begin(115200);
    wifi.disconnect(true);
    wifi.begin(ssid,password);
    Serial.println("WiFi 正在连接");
    While(WiFi.status() != WL_CONNECTED);
    {
        Serial.print("WiFi 正在连接");
        digitalWrite(LED, LOW);
        delay(500);
        digitalWrite(LED, HIGH);
        delay(500);
    }
}

```

```
}

Serial.println("WiFi 已连接");
Serial.print("IP 地址: ");
Serial.println(WiFi.localIP());

}

void loop()
{
    digitalWrite(LED, HIGH);
}
```

