

绝密★启用前

2020 年 10 月高等教育自学考试全国统一命题考试

微型计算机原理与接口技术

(课程代码 02205)

注意事项:

1. 本试卷分为两部分, 第一部分为选择题, 第二部分为非选择题。
2. 应考者必须按试题顺序在答题卡(纸)指定位置上作答, 答在试卷上无效。
3. 涂写部分、画图部分必须使用 2B 铅笔, 书写部分必须使用黑色字迹签字笔。

第一部分 选择题

一、单项选择题: 本大题共 15 小题, 每小题 1 分, 共 15 分。在每小题列出的备选项中只有一项是符合题目要求的, 请将其选出。

1. 冯·诺依曼结构计算机的 ALU 称为
A. 控制器 B. 比较器
C. 加法器 D. 运算器
2. 若两个 8 位二进制数 10000011B 与 01011110B 的运算结果是 11011101B, 则该运算是
A. 二进制“加” B. 二进制“与”
C. 二进制“异或” D. 二进制“减”
3. 已知字符'A'的 ASCII 码值是 65, 执行语句“char c='C'; printf("%d,%c",c,c+3);”的输出是
A. 67, F B. C, F
C. 67, 70 D. C, 70
4. 设有 C 语言定义: “int m=5, n=1;”, 下列选项中 n 值与其它不同的表达式是
A. n*=m/2 B. n=(++m)/3
C. n+=m%2 D. n=(m--)-2

5. 下列选项中, 与 C 语言语句“y=!x?~x:x;”功能相同的是
A. if(x!=0) y=x; else y=~x; B. if(x==0) y=x; else y=~x;
C. if(!x) y=x; else y=~x; D. if(x) y=~x; else y=x;
6. 在 C 语言程序中, 生存期从程序运行开始一直持续到程序运行结束的变量是
A. 自动变量和外部变量 B. 内部静态变量和外部变量
C. 实参变量和形参变量 D. 自动变量和外部静态变量
7. LPC2138 微控制器正在执行的指令的地址是
A. PC 值 B. PC-4 值
C. PC-8 值 D. PC+4 值
8. ARM 处理器的 LR 寄存器用于存放
A. 堆栈地址 B. 主程序地址
C. 当前程序状态和控制位 D. 子程序的返回地址
9. LPC2138 微控制器执行如下运算操作: 0xFFA00FF0+0x0A00FF00, 则条件标志位 Z、V 分别为
A. 0、0 B. 0、1
C. 1、0 D. 1、1
10. LPC2138 微控制器的片内 32KB SRAM 可以用来存储
A. 中断向量表 B. 代码
C. 通用寄存器 D. 外设地址
11. LPC2138 微控制器支持存储器重映射功能, 指的是
A. 对被映射单元的内容进行复制
B. 对被编址的存储器地址重新更换为新的物理地址
C. 将多个逻辑地址指向了同一个物理存储单元
D. 将多个物理存储单元指向了同一个逻辑地址
12. 在 ARM 系统中, 程序中不会产生堆栈操作的情况是
A. 函数调用 B. 异常发生
C. 中断处理 D. 程序跳转
13. 在 ARM 处理器中, 一般通用寄存器的数量是
A. 4 个 B. 8 个
C. 12 个 D. 16 个
14. LPC2138 芯片的三个引脚功能选择寄存器都是
A. 4 位 B. 8 位
C. 16 位 D. 32 位

15. 异步传输数据帧格式中不包含的信息位为

- A. 空闲位
- B. 起始位
- C. 停止位
- D. 数据位

第二部分 非选择题

二、填空题：本大题共 14 小题，每小题 2 分，共 28 分。

16. C 语言程序中由 if-else 语句构成的结构称为_____结构，由 for 或 while 语句构成的结构称为_____结构，它们都是程序设计时常用的结构。

17. 若有定义：“unsigned char x = 0xF4;”，则表达式 (x>>4) & (~0<<3) 的值是十六进制数_____；将 x 的高 4 位置 0 的表达式是_____。

18. 设有定义：“enum color{red, orange=10, yellow, green, cyan, blue, violet} c, d;”，若执行了“c=red; d=green;”，则语句“printf(“%d, %d”, c, d);”的输出结果是_____。

19. 补充完整下面用递归函数求 $\text{Fun}(n) = n \times \text{Fun}(n-1) + 1$, (n 为非负整数) 的 C 函数：

```
long fun(_____)
{
    if (n== 0) return 1;
    if (n>0) return _____;
}
```

20. 若有定义：“char str[] = “code”, *p = str;”，则数组 str 共有_____个数组元素，执行语句“printf(“%c”, *(p + (*p - 'a')));”的输出结果是_____。

21. 设有三个整型变量 y、m、d，y 需要 6 位二进制数，m 需要 4 位二进制数，d 需要 5 位二进制数。若要用 2 个字节保存这三个变量的值，可以采用_____，相应的定义语句为：“struct packe_ymd {_____}p;”。

22. LPC2138 微控制器的 ARM 内核包含_____个 CPSR 和_____个专供异常模式使用的 SPSR。

23. LPC2138 芯片片内 Flash 通过_____位宽度的 ARM7 局部总线与 ARM7TDMI-S CPU 相连。

24. 按照小端字节顺序将 0x12345678 存入地址 0x00000010 中，则地址 0x00000011 中存入的数据为_____，地址 0x00000012 中存入的数据为_____。

25. LPC2138 微控制器的 CPU 时钟由晶振时钟或外部时钟经 PLL 模块_____得到，时钟 PCLK 经过 VPB 模块_____得到，供外设器件使用。

26. RS-232C 电缆的长度不超过_____米。

27. 当 ARM 处理器的数据访问指令的_____不存在或者该地址不允许访问，处理器产生数据访问中止异常中断。

28. VIC 是_____和 ARM 内核之间的桥梁。

29. 中断过程包含中断响应、_____、_____三个阶段。

三、改错题：本大题共 4 小题，每小题 2 分，共 8 分。每小题只有一处错误或不妥，请指出，并将其改正。

30. #include <stdio.h>

```
int main(void)
{
    const float area = 3.14;
    float r = 33.78;
    area *= r * r;
    printf("Area is %7.2f\n", area);
}
```

31. 下述程序用于计算 $s = 1 \times 3 \times 5 \times \dots \times (2N-1)$ 。指出程序中的错误并改正：

```
#include <stdio.h>
#define N 50
int main(void)
{
    short int k = 1;
    long int s = 1;
    for (k = 1; k < N; k++)
        s *= 2*k - 1;
    printf("result is %ld\n", s);
}
```

32. 以下函数用于求三个浮点数中的最小数，指出下述程序中的错误并改正：

```
float minimum(float x, float y, float z)
{
    float min = x < y ? x < z ? x : (y < z ? y : z);
    return min;
}
```

33. struct os {

```
    char name[20];
    float version;
} mobile = {"Android", 7.0};
```

```

int main(void)
{
    struct os *p = &mobile;
    float *pv = p->version;
    printf("The OS is %s v. %f\n", p->name, *pv);
}

```

四、程序阅读题：本大题共 4 小题，每小题 4 分，共 16 分。

34. #include <stdio.h>

```

int main(void)
{
    int i = 0;
    while (i < 4) {
        switch (i + 'A') {
            case 'A': printf("%d->A,", i++); break;
            case 'B': printf("%d->B,", i++);
            default: printf("%d->D,", ++i);
        }
    }
}

```

程序运行结果为：

35. #include <stdio.h>

```

void inorout (int x, int *y)
{
    static int h = 0;
    if (x > 0) (*y)++;
    else (*y)--;
    printf("%d, ", ++h);
}

int main(void)
{
    int h = 3, t = 5;
    inorout (h, &t);
    inorout (-1, &h);
    printf("%d, %d\n", h, t);
}

```

程序运行结果为：

36. #include <stdio.h>

```

int main(void)
{
    char str[] = "aBeD";
    char *p;
    for (p = str; p < str + 4; p++) {
        if (*p >= 'a' && *p <= 'z')
            *p = *p - 'a' + 'A';
        else
            *p = *p - 'A' + '0';
        printf("%s\t", p);
    }
}

```

程序运行结果为：

37. #include <stdio.h>

```

typedef struct list {
    int data;
    struct list *next;
} st_list;

int main(void)
{
    st_list line[4], *p = line;
    for (int i = 0; i < 3; i++, p++) {
        p->data = i * i + 1; p->next = p + 1;
    }
    p->next = p - 1;
    p = p - 2;
    p->next = p->next->next;
    (p+1)->next = 0;
    p->next->data = 10;
    for (p = line; p != 0; p = p->next)
        printf("%d, ", p->data);
}

```

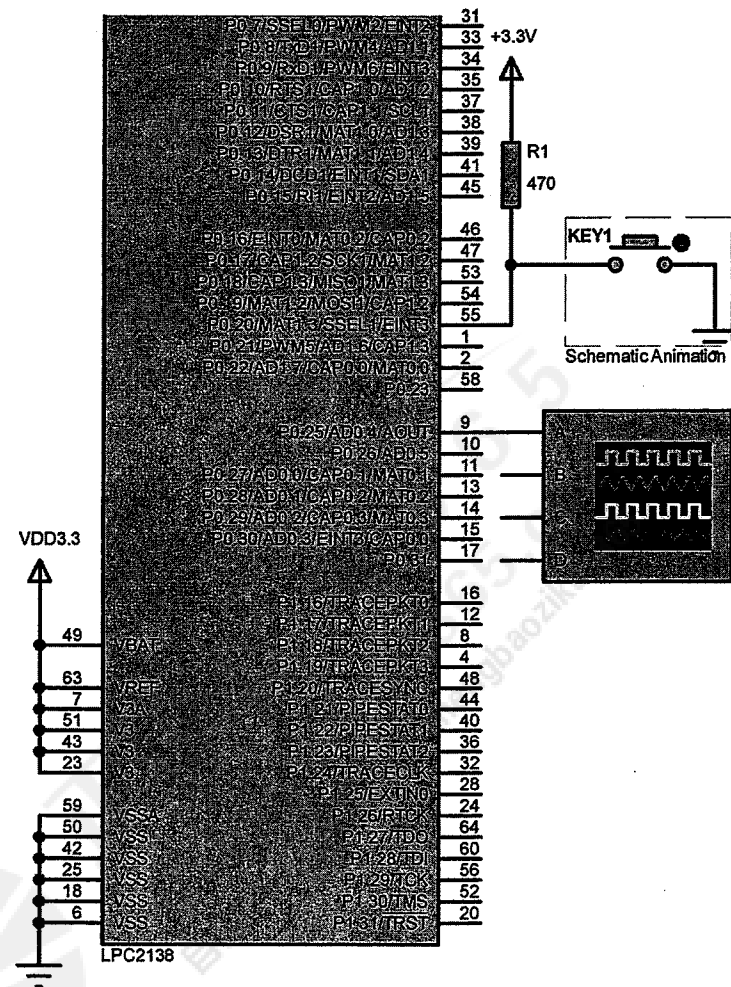
程序运行结果为：

五、程序设计题：本大题共 1 小题，共 8 分。

38. 编写一个函数 charCount，计算一个字符串中给定英文字母出现的次数（不区分大小写）。例如：给定字符串"CongrAduation"，给定字母'a'，则计算'A'和'a'在字符串"CongrAduation"中出现的总次数，调用 charCount("CongrAduation",'a')，返回值为 2。

六、分析题：本大题共 2 小题，每小题 10 分，共 20 分。

39. 题 39 图（1）所示为由 LPC2138 微控制器构成的产生锯齿波和三角波的电路，试阅读程序，回答问题，将编号①~④处空缺的内容填写在答题卡上，在答题卡上完成题 39 图（2）和题 39 图（3）的作图。



题 39 图（1）

```
/* Main.c */
#include <LPC2138.h>
typedef unsigned int  uint32;
typedef unsigned short uint16;
```

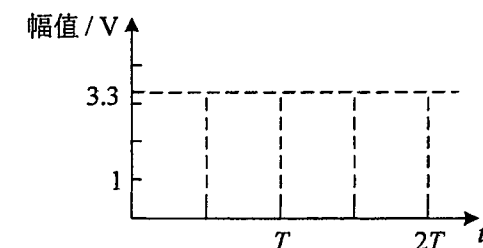
```
typedef unsigned char  uint8;
#define KEY1 (IO0PIN&(1<<20))
uint16 staus=1;
/* 名称: DelayNS() */
void DelayNS(uint32 dly)
{
    uint32 i;
    for(; dly>0; dly--)
        for(i=0; i<50000; i++);
}
/* 名称: juchi() */
/* 功能: 产生锯齿波 */
void juchi(void)
{
    uint16 j,i;
    for(i=622;i<1024;i++)
    {
        DACR=(i<<6)|(1<<16);
        for (j=0; j<0x20; j++);
    }
}
/* 名称: TriAngle() */
/* 功能: 产生三角波 */
void TriAngle(void)
{
    uint16 j,i=0;
    for(i=0;i<623;i++)
    {
        DACR=(i<<6)|(1<<16);
        for (j=0; j<0x20; j++);
    }
    for(i=622;i>0;i--)
    {
        DACR=(i<<6)|(1<<16);
```

```

        for (j=0; j<0x20; j++);
    }
}
/* 名称: KeyScan() */
void KeyScan(void)
{
    if(!KEY1)
    {
        DelayNS(1);      /*延时去抖动*/
        if(!KEY1)
        {
            staus++;
            if(staus>1)
                staus=0;
        }
        while(!KEY1);    /*等待键松开*/
    }
}
/* 名称: main() */
int main(void)
{
    IO0DIR = IO0DIR & ~(0x01 << 20);
    PINSEL1 = (PINSEL1 & ~(0x03 << 18)) | (0x02 << 18); /* P0.25 连接 Aout */
    while(1)
    {
        KeyScan();
        switch(staus)
        {
            case 0: TriAngle(); break; /*产生三角波*/
            case 1: juchi(); break;   /*产生锯齿波*/
            default: break;
        }
    }
}

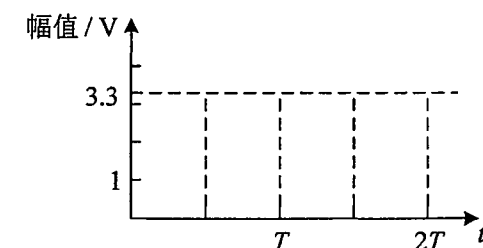
```

- (1) 程序运行后, P0.25 引脚输出 ① (锯齿/三角) 波; 若点按按键 KEY1 一次后, P0.25 引脚输出 ② (锯齿/三角) 波。
- (2) 如果要求 P0.25 引脚输出锯齿波形的周期增加 1 倍, 则应将函数 juchi() 中的语句 ③ 修改为 ④。
- (3) 在题 39 图 (2) 中画出锯齿波的波形 (画出 2 个周期波形, 假设周期为 T)。



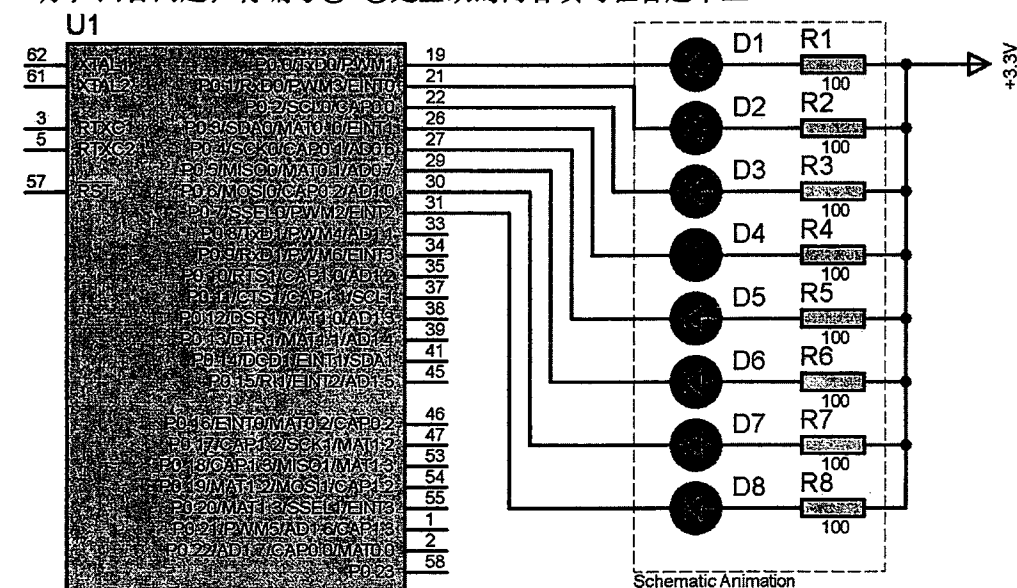
题 39 图 (2)

- (4) 在题 39 图 (3) 中画出三角波的波形 (画出 2 个周期波形, 假设周期为 T)。



题 39 图 (3)

40. 题 40 图所示为由 LPC2138 微控制器和 D1~D8 发光二极管构成的电路, 试阅读程序, 回答问题, 将编号①~⑩处空缺的内容填写在答题卡上。



题 40 图

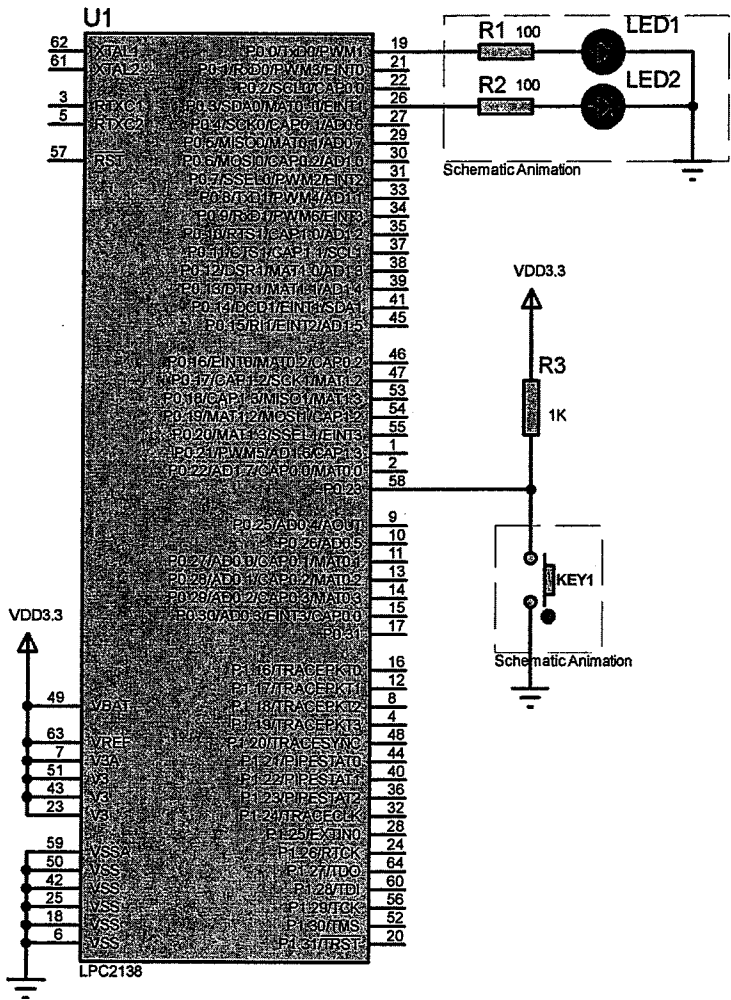
```
/* Main.c */
#include <LPC2138.h>
unsigned long LED_ON = 0x80;
unsigned char LED_ON_Num = 8;
void delay_ms(unsigned int t)
{
    const unsigned int cclk = 60000000;
    const unsigned int loop_clks = 10;
    volatile unsigned int n;
    n = (cclk/1000/loop_clks) * t;
    while(n--);
    return;
}
int main()
{
    IO0DIR = 0x000000ff;
    while(1)
    {
        IO0SET = ~LED_ON;
        IO0CLR = LED_ON;
        LED_ON >>= 1;
        delay_ms(200);
        if (LED_ON_Num-- == 2)
        {
            LED_ON = 0x80;
            LED_ON_Num = 8;
        }
    }
}
```

- (1) 程序运行后，观察到的现象为：发光二极管轮流点亮，每一时刻点亮 ① 个发光二极管，首先点亮发光二极管 ②，其次点亮发光二极管 ③，最后点亮发光二极管 ④，然后循环往复。
- (2) 第一次执行语句 “IO0SET = ~LED_ON;” 后，点亮的发光二极管为 ⑤，熄灭的发光二极管为 ⑥。

- (3) 第一次执行语句 “IO0CLR = LED_ON;” 后，点亮的发光二极管为 ⑦，熄灭的发光二极管为 ⑧。
- (4) 语句 “IO0DIR = 0x000000ff;” 的作用是将引脚 ⑨ (填写引脚符号) 设置为 ⑩ 方向。

七、应用题：本大题共 1 小题，共 5 分。

41. 题 41 图所示为由 LPC2138 微控制器构成的按键及发光二极管电路，要求实现如下功能：每点按一次按键 KEY1，则 LED1、LED2 的状态取反。试完善程序，将编号 ①~⑤ 处空缺的内容填写在答题卡上。



题 41 图

```
/* Main.c */
#include <LPC2138.h>
```

```

typedef unsigned int uint32;          /* 无符号 32 位整型变量 */
#define keycon ①
#define KEY1 (1<<23) /*定义按键*/
#define LED1 (1<<0)  /*定义 LED 引脚*/
#define LED2 (1<<3)
/* 函数名称: keyscan() */
/* 函数功能: 按键扫描, 每按一次键对相应的 LED 进行取反 */
uint32 keyscan(void)
{
    uint32 j,k,key;
    k=IO0PIN;
    if((k&keycon)!=keycon)
    {
        for(j=0;j<50000;j++); /*延时去抖动*/
        k=IO0PIN;
        if((k&keycon)!=keycon) key=(uint32) KEY1;
        while((k&keycon)!=keycon) k=IO0PIN;
    }
    else
        key=0x0;
    return key;
}
/* 函数名称: main() */
int main(void)
{
    uint32 key;
    PINSEL0 = 0x00000000;
    PINSEL1 = 0x00000000;
    ② = 0x000000ff;
    IO0CLR = 0xff;
    IO0SET =0xff;
    while(1)
    {
        key=keyscan();

```

```

if (key!=0x ③)
{
    ④ = IO0PIN ^ LED1;
    ⑤ = IO0PIN ^ LED2;
}
}
}

```