

2022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最符合题目要求的, 请将其选出。

1. 被称为“计算机之父”的是
A. 图灵
B. 冯·诺伊曼
C. 比尔·盖茨
D. 乔布斯
2. 某计算机系统中整数用 8 位补码表示, 则 $10010100B + 00101111B$ 结果的十进制表示是
A. -67
B. -61
C. 67
D. 195
3. 语句“`printf("%X", -1);`”的正确输出是
A. -1
B. -0x1
C. 0xFFFFFFFF
D. FFFFFFFF
4. 如有定义:“`struct {char s[6]; int a;} obj, *p = &obj;`”, 则下列选项中, 非法的表达式是
A. `*p.s[2]`
B. `p->a`
C. `obj.s[4]`
D. `p->s[2]`

5. 执行下面的语句后, s 的值是

```

s = 10;
k = 'a';
while (k <= 'f') {
    for (j = 0; j < k - 'a'; ++j)
        ++s;
    k += 2;
}

```

- | | |
|-------|-------|
| A. 12 | B. 14 |
| C. 16 | D. 18 |

6. 下面循环中循环体执行次数是

```

for (k = 1; k < 100; ++k)
    if (k % 5 == 0) break;

```

- | | |
|-------|--------|
| A. 4 | B. 5 |
| C. 99 | D. 100 |

7. 所有 ARM 内核采用的体系结构是

- | | |
|------------|---------|
| A. STM32 | B. CISC |
| C. LPC2138 | D. RISC |

8. ARM7TDMI 采用的指令流水线为

- | | |
|------------|------------|
| A. 一级指令流水线 | B. 两级指令流水线 |
| C. 三级指令流水线 | D. 四级指令流水线 |

9. ARM 处理器运行模式中, 特权模式数量有

- | | |
|--------|--------|
| A. 4 种 | B. 5 种 |
| C. 6 种 | D. 7 种 |

10. ARM 处理器中, 堆栈指针寄存器 (SP) 用于保存

- | | |
|---------|---------|
| A. 栈顶地址 | B. 栈底地址 |
| C. 栈顶数据 | D. 栈底数据 |

11. 在 LPC2138 微控制器中, 可以作为一个字的起始存储地址的是

- | | |
|---------------|---------------|
| A. 0x4000600A | B. 0x4000600B |
| C. 0x4000600C | D. 0x4000600D |

12. LPC2138 微控制器复位后, 所有 GPIO 引脚默认状态为

- | | |
|------------|---------|
| A. 输出低电平状态 | B. 高阻状态 |
| C. 输出高电平状态 | D. 输入状态 |

13. LPC2138 的向量中断控制器 (VIC) 专门管理
- A. 中断响应 B. 中断返回
C. 中断服务 D. 中断请求
14. 比特率的单位通常是
- A. kbit/s B. Byte/s
C. Word/s D. KByte/s
15. 一个 8 位 D/A 转换器, 若满量程电压为 5V, 其分辨率约为
- A. 0.0039 B. 0.0196
C. 0.125 D. 0.6250

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. 编译阶段生成与源程序等效的_____代码或_____代码。
17. 如有定义: “int a[] = {2,5,1,7,8};”, 则数组 a 有_____个元素, a[2]的值是_____。
18. 如果 m=5, n=0, 则表达式 (m == 5) && (n += 4) 的值是_____, 变量 n 的值是_____。
19. 函数 f 的功能是返回本函数被调用的次数, 请填写完成函数 f 的定义。

```
int f()
{
    _____ int count = 0;
    return _____;
}
```

20. 函数 count 统计字符串中满足 g 函数中指定条件的字符数。如 g 函数定义为

```
int g(char c) {return c=='a';}
```

调用 count(“abcabcabc”)的返回值是 3。请填写完成函数 count 的定义。

```
int count(const char *s)
{
    int k=0;
    for (; _____; ++s)
        if ( _____ ) ++k;
    return k;
}
```

21. 某个栈的结点类型 Node 定义如下:

```
struct Node {
    int data;
    struct Node *next;
};
```

栈顶指针是 stack。此外, 系统中所有空闲单元也被链成一个链接栈, 栈顶指针为 Free。stack 和 Free 都是外部变量。以下是进栈函数的实现, 请填空。

```
int push(int data)
{
    struct Node *p = Free;
    if (Free == NULL) return 0; /* 没有空闲空间, 进栈失败 */
    Free = _____;
    p->data = data;
    p->next = _____;
    stack = p;
    return 1;
}
```

22. 指令只能通过 PC 给定的地址从_____读入 CPU 并执行。
23. ARM 处理器运行模式中, 特权模式又分为系统模式和_____。
24. 在 ARM 体系的存储器中, 按照小端字节顺序将数据 0x12345678 存入地址 0x40001000 中, 则地址 0x40001002 中存入的字节数据为十六进制数_____。
25. LPC2138 微控制器的 GPIO 寄存器中, IODIR 寄存器控制各个引脚的_____。
26. LPC2138 微控制器的引脚设置为 GPIO 工作方式时, 指令 IOODIR = 0x00000080; 将引脚_____配置为输出。
27. 当处理器执行完中断服务程序后, 返回到先前程序继续执行的过程称为_____。
28. 异步串行通信中, 只允许单一方向的数据传输, 称为_____通信。
29. D/A 转换器的绝对精度是指 D/A 实际输出电压与_____之间的误差。

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. 某程序输出小写字母 a 到 z 的 ASCII 值的十进制表示。指出程序中的错误并改正。

```
int main()
{
    char ch;
    for (ch = 'a'; ch <= 'z'; ++ch) printf("%c\n", ch);
    return 0;
}
```

31. 某程序员编写了一个统计整型数组中能被 5 整除的元素个数的函数。指出其中的错误并改正。

```
int count(int data[], int size)
{
    int k = 0, i;
    for (i = 0; i < size; ++i)
        if (data[i] = data[i] / 5 * 5) ++k;
    return k;
}
```

32. 设 c2、c4、i 和 size 都是整型变量，a 是整型数组，以下程序段统计有 size 个元素的数组 a 中能被 2、4 整除的元素个数，c2 是能被 2 整除的元素个数，c4 是能被 4 整除的元素个数，指出其中的错误并改正。

```
for (c2 = c4 = i = 0; i < size; ++i)
    switch (a[i] % 4) {
        case 0: ++c4; break;
        case 2: ++c2; break;
    }
```

33. 指出下列程序中的错误并改正。

```
int f(int a[], int size)
{
    const int s = 0, k;
    for (k = 0; k < size; k += 3)
        s += a[k];
    return s;
}
```

四、程序阅读题：本大题共 4 小题，每小题 4 分，共 16 分。

34. 写出下列程序的执行结果。

```
#include <stdio.h>
int main()
{
    int a[] = {2, 5, 3, 7};
    int ch;
    for (ch = 0; ch < 4; ++ch)
    {
        while (--a[ch])
            switch (ch) {
                case 0: printf("%c", a[ch] + 'a'); break;
                case 1: printf("%c", a[ch] + 'b'); break;
            }
    }
}
```

```
case 2: printf("%c", a[ch] + 'A'); break;
default: printf("%c", a[ch] + '0'); break;
```

```
}
printf("\n");
```

```
}
return 0;
```

```
}
```

35. 写出下列程序的执行结果。

```
#include <stdio.h>
void fun1(int n);
void fun2();
int glob = 5;
int main()
{
    int k1;
    for (k1 = 0; k1 < 2; ++k1) {
        fun1(k1);
        fun2();
    }
    return 0;
}
void fun1(int n) { printf("%d\t", n+glob); }
void fun2() { printf("%d\t", ++glob); }
```

36. 写出下列程序的执行结果。

```
#include <stdio.h>
void f(char *s);
int main()
{
    char str[] = "abcdefg123";
    f(str);
    f(str+3);
    return 0;
}
void f(char *p)
{
    printf("%c\t", *p);
    p[5] = 'A';
    printf("%s\t", p);
}
```

37. 写出下列程序的执行结果。

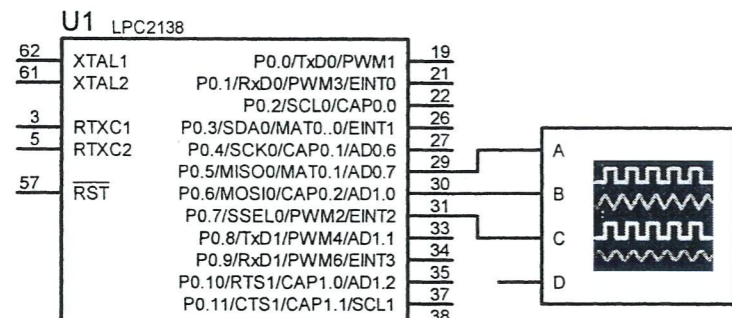
```
#include <stdio.h>
union sample {
    unsigned char a;
    struct {
        unsigned char a1:3;
        unsigned char a2:2;
        unsigned char a3:3;
    } b;
};
int main()
{
    union sample test;
    test.a = 0x1F;
    printf("%o\t%o\t%o\t",--test.b.a1,test.b.a2,++test.b.a3);
    printf("%x \n",test.a);
    return 0;
}
```

五、程序设计题：本大题共 1 小题，共 8 分。

38. 编写一个函数 max，参数是一个有 n 个元素的整型数组，元素值是整数 0~9，函数 max 返回由这 n 个数字组成的最大的 n 位数（每个数字仅用 1 次）。例如：有定义数组 “int a[] = { 0, 8, 5, 0, 7, 7 }”，数组 a 有 6 个元素，调用函数 max(a,6)，返回值是整数 877500。

六、分析题：本大题共 2 小题，每小题 10 分，共 20 分。

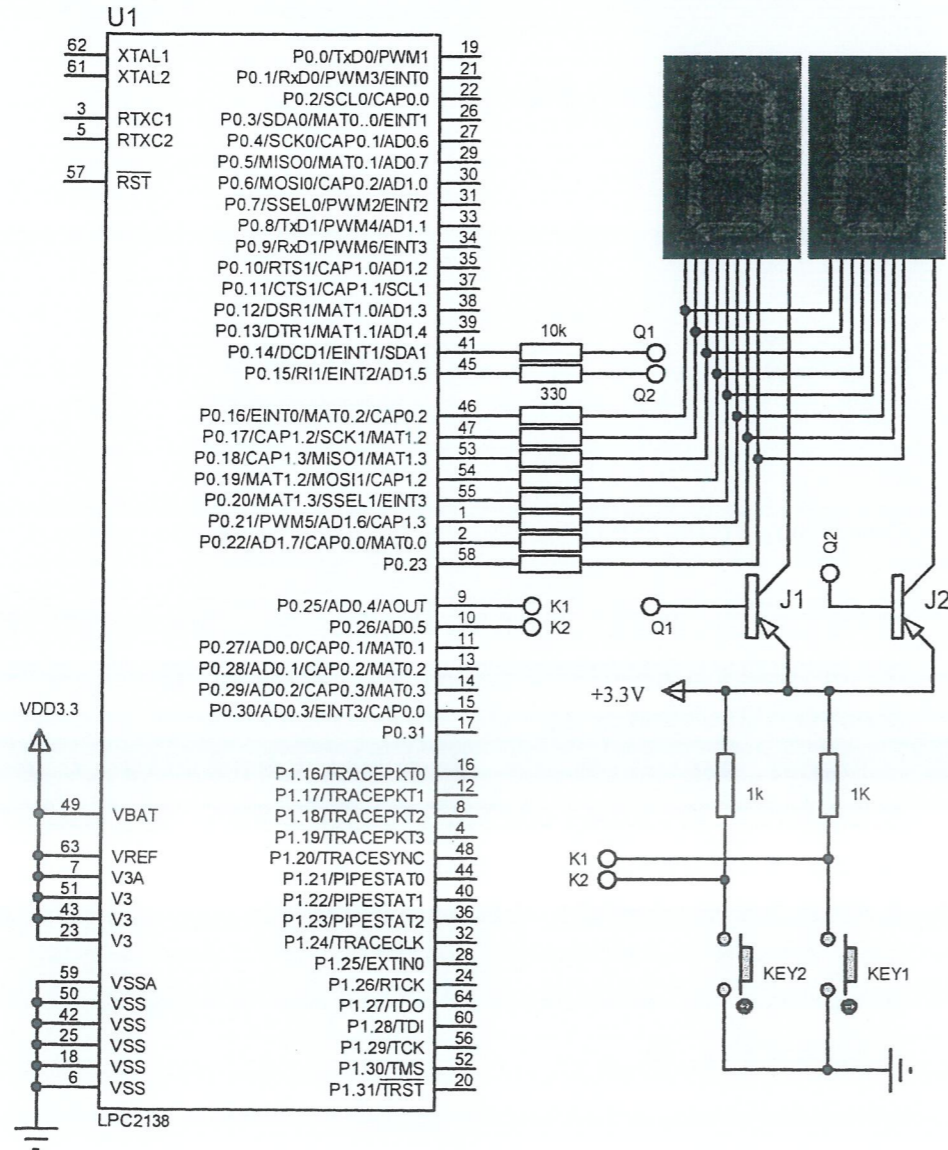
39. 如题 39 图所示为 LPC2138 微控制器电路，用来输出不同占空比的矩形波形。电路的晶振频率为 12MHz。试阅读下述程序，回答问题，将编号①~⑩处空缺的内容填写在答题卡上。



题 39 图

```
/* Main.c */
#include <LPC2138.h>
typedef unsigned int uint32; // 无符号 32 位整型变量
#define POUT (uint32)((1 << 5)|(1 << 6)|(1 << 7))
uint32 m_1ms_count;
void __irq IRQ_Timer0 (void) // 定时器 0 中断服务程序
{
    m_1ms_count++;
    m_1ms_count %= 1000;
    switch(m_1ms_count) {
        case 0: IO0SET = POUT; break;
        case 333: IO0CLR = (1<<5); break;
        case 500: IO0CLR = (1<<6); break;
        case 667: IO0CLR = (1<<7); break;
        default: break;
    }
    T0IR = 0x01;
    VICVectAddr = 0x00;
}
void Time0Init(void)
{
    T0TC = 0; // 定时器 T0 初始化
    T0PR = 99; // 设置定时器 0 分频为 100 分频，得 150000Hz
    T0MCR = 0x03; // 匹配通道 MR1 匹配中断并复位 TC
    T0MR0 = 150; // MR0 比较值
    T0TCR = 0x03; // 启动并复位 T0TC
    T0TCR = 0x01; // 使能定时器 0
    T0IR = 0x01;
    VICIntSelect = 0x00; // 所有中断通道设置为 IRQ 中断
    VICVectCntl0 = 0x20 | 0x04; // 分配定时器 0 中断到通道 0 (最高优先级)
    VICVectAddr0 = (uint32) IRQ_Timer0; // 设置中断服务程序地址向量
    VICIntEnable = (1<<4); // 使能定时器 0 中断
}
int main(void)
{
    PINSEL0 = 0x00;
    IO0DIR = POUT;
    Time0Init();
    IO0CLR = POUT;
    while(1);
}
```

- (1) 主程序中, 语句“PINSEL0 = 0x00;”的作用是将引脚__①__设置为__②__功能。
 语句“IO0DIR = POUT;”的作用是将引脚__③__的方向设置为__④__。
- (2) 程序运行后定时器__⑤__ (0/1) 每隔__⑥__ms 产生一次中断。
- (3) 程序运行后 P0.5 引脚输出频率为__⑦__Hz、占空比为__⑧__的矩形波; P0.6 引脚输出波形的占空比为__⑨__; P0.7 引脚输出波形的占空比为__⑩__。
40. 如题 40 图所示为 LPC2138 微控制器构成的键控数码管电路, 按键按下后立即释放。试阅读下述程序, 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>
typedef unsigned long uint32_t;
typedef unsigned char uint8_t;
typedef union{
    uint32_t data;
    struct {
        uint32_t b0:8, b1:8, b2:8, b3:8;
    } bytes; // 按字节操作
    struct {
        uint32_t d0:1, d1:1, d2:1, d3:1, d4:1, d5:1, d6:1, d7:1;
        uint32_t d8:1, d9:1, d10:1, d11:1, d12:1, d13:1, d14:1, d15:1;
        uint32_t d16:1, d17:1, d18:1, d19:1, d20:1, d21:1, d22:1, d23:1;
        uint32_t d24:1, d25:1, d26:1, d27:1, d28:1, d29:1, d30:1, d31:1;
    } field; // 按位操作
} reg_bits_t, reg_io_pin_t, reg_io_set_t, reg_io_clr_t, reg_io_dir_t;
#define rIO0PIN (*(volatile reg_io_pin_t *) 0xE0028000)
#define rIO0SET (*(volatile reg_io_set_t *) 0xE0028004)
#define rIO0DIR (*(volatile reg_io_dir_t *) 0xE0028008)
#define rIO0CLR (*(volatile reg_io_clr_t *) 0xE002800C)
#define Q1 14 // 控制数码管的十位
#define Q2 15 // 控制数码管的个位
#define gpio_bit(reg, g, i) rIO##g##reg.field.d##i
#define gpio_byt(reg, g, i) rIO##g##reg.bytes.b##i
#define set_seg(code) gpio_byt(PIN, 0, 2) = code;
#define lighten(bit) gpio_bit(CLR, 0, bit) = 1;
#define darken(bit) gpio_bit(SET, 0, bit) = 1;
uint8_t disp_code[] = {0xc0, 0xf9, 0xa4, 0xb0, 0x99, 0x92, 0x82, 0xf8, 0x80, 0x90};
// 数字 0~9 的显示码
void DelayMs(unsigned int t); // 延时函数, 延时 t ms。
int IsKeyPressed(volatile int key_state);
void InitRegs(void); // 寄存器初始化, 设置引脚 P0.14~P0.23 为 GPIO 输出;
// 设置引脚 P0.25~P0.26 为 GPIO 输入。
void DisplayNum(int n);

```

```

int main(void)
{
    int n = 0, up_down=0;
    InitRegs();
    darken(Q2);
    darken(Q1);
    while(1)
    {
        if(IsKeyPressed(gpio_bit(PIN,0,25))) // 检查按键
        {
            if (up_down == 0) n++;
            else n--;
            if (n == 10) up_down = 1;
            if (n == 0) up_down = 0;
        }
        DisplayNum(n); // 显示计数值
    }
}

```

```

void DisplayNum(int n) // 显示计数器的个位和十位
{
    lighten(Q2);
    darken(Q1);
    set_seg(disg_code[n%10]);
    DelayMs(50);
    darken(Q2);
    lighten(Q1);
    set_seg(disg_code[n/10]);
    DelayMs(50);
}

```

```

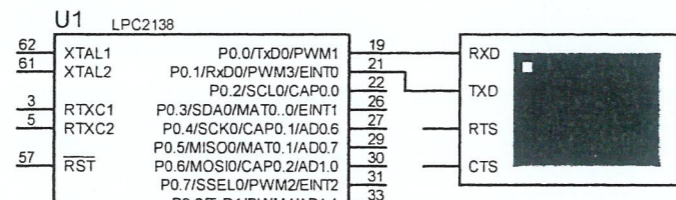
int IsKeyPressed(volatile int key_state)
{
    static signed char key_st = -1;
    if(key_state == 0) // KEY1 按下
    {
        DelayMs(5);
        if(key_state == 0)
            key_st <<= 1;
    }
    else // KEY1 释放
    {
        DelayMs(5);
        if(key_state != 0)
        {
            key_st <<= 1;
            key_st |= 1;
        }
    }
    return ((key_st & 0x3) == 1);
}

```

- (1) 程序运行后，两个数码管从左到右显示的数值为__①__，接着点按 KEY1 键一次，两个数码管从左到右显示的数值为__②__，点按 KEY2 键一次，两个数码管从左到右显示的数值为__③__。
- (2) 程序运行后，点按 KEY1 键 10 次，两个数码管从左到右显示的数值为__④__，接着点按 KEY1 键 5 次，两个数码管从左到右显示的数值为__⑤__，点按 KEY2 键一次，两个数码管从左到右显示的数值为__⑥__。
- (3) 首次执行函数 IsKeyPressed()，变量 key_st 的初值为 0x__⑦__。当 KEY1 未按下时，执行函数 IsKeyPressed()后返回值等于 0x__⑧__，此时，变量 key_st 的最低两位等于__⑨__B；当 KEY1 按下后立即释放时，执行函数 IsKeyPressed()后返回值等于 0x__⑩__。

七、应用题：本大题共 1 小题，共 5 分。

41. 如题 41 图所示为 LPC2138 微控制器的 UART 通信电路，使用 UART0 收发数据，波特率为 9600，8 位数据位，1 位停止位，无奇偶校验。线路状态寄存器 U0LSR/U1LSR 各位的含义见题 41 表所示。试完善下述程序。将编号①~⑤处空缺的内容填写在答题卡上。



题 41 图

题 41 表 线路状态寄存器 U0LSR/U1LSR 各位的含义

位	功能	描述
0	接收数据就绪	0: 接收数据寄存器为空
		1: 接收数据寄存器中有未读取的数据
1	溢出错误	0: 未发生溢出错误
		1: 发生溢出错误
2	奇偶校验错误	0: 未发生奇偶校验错误
		1: 发生奇偶校验错误
3	帧错误	0: 未发生帧错误
		1: 发生帧错误
4	间隔中断	0: 未发生间隔中断
		1: 发生间隔中断
5	发送保持寄存器空	0: 发送保持寄存器中有数据
		1: 发送保持寄存器为空
6	发送器空	0: 发送保持寄存器或发送位移寄存器中有数据
		1: 发送保持寄存器和发送位移寄存器为空
7	RX FIFO 错误	0: 接收过程中没有发生 RX 错误
		1: 接收过程中发生至少一个 RX 错误

```

/* Main.c */
#include <LPC2138.h>
typedef unsigned char uint8;
void InitUART0()
{
    U0LCR = 0x80;
    U0DLL = 72;
    U0DLM = 0;
    U0LCR = 0x03;
}

void SendByte(uint8 data)
{
    U0THR = data;
    while((U0LSR & 0x__①__) == __②__);
}

uint8 GetByte()
{
    uint8 data;
    while((U0LSR & 0x__③__) == __④__);
    data = U0RBR;
    return data;
}

int main()
{
    uint8 data;
    PINSEL0 = 0x00000005;
    __⑤__;
    while(1)
    {
        data = GetByte();
        SendByte(data);
    }
}

```