

2006 年下半年嵌入式系统设计师下午试题

(考试时间 14 : 00~16 : 30 共 150 分钟)

请按下述要求正确填写答题纸

- 1. 在答题纸的指定位置填写你所在的省、自治区、直辖市、计划单列市的名称。
- 2. 在答题纸的指定位置填写准考证号、出生年月日和姓名。
- 3. 答题纸上除填写上述内容外，只能填写解答。
- 4. 本试卷的五道试题都是必答题，每道试题均为 15 分，满分 75 分。
- 5. 解答时字迹务必清楚，字迹不清时，将不评分。
- 6. 仿照下面例题，将解答写在答题纸的对应栏内。

例题

2006 年下半年全国计算机技术与软件专业技术资格(水平)考试日期是 (1) 月 (2) 日。
因为正确的解答是“11 月 4 日”，故在答题纸的对应栏内写上“11”和“4” (参看下表)。

例题	解答栏
(1)	11
(2)	4

试题一 (15 分)

阅读下面有关车载 GPS 终端系统的叙述，回答问题 1 至问题 3，将答案填入答题纸的对应栏中。

【说明】

GPS 是全球定位系统 (Global Position System)。车载 GPS 终端是置于机动车内的实时定位装置，它的应用对象是需要定位和调度的车辆。车辆可以通过终端与 GPS 进行实时、准确的定位，并能够通过无线通讯网络上报远程的车辆调度中心。中心可以通过 终端远程监视车行轨迹，并可在特殊情况下通过终端控制车辆。同时，终端还装备车载电话，可以在出现特殊情况时及时地通知车辆调度中心。图 1-1 所示为车载终端系统的用例图，对于车载 GPS 终端系统来说，主要的角色有两个：车辆调度中心用户和车载终端用户。

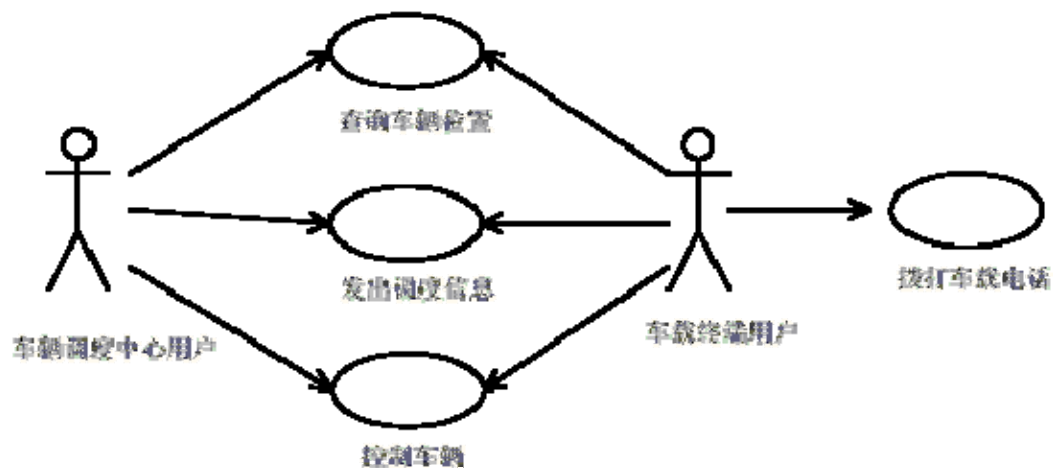


图 1-1 车载终端系统用例

图 1-2 所示为车载终端系统中的 GSM 无线电通讯模块的状态图，用于与调度中心进行联系。GSM 模块共有四个状态：通话中、有问题、待命、短消息通讯中。

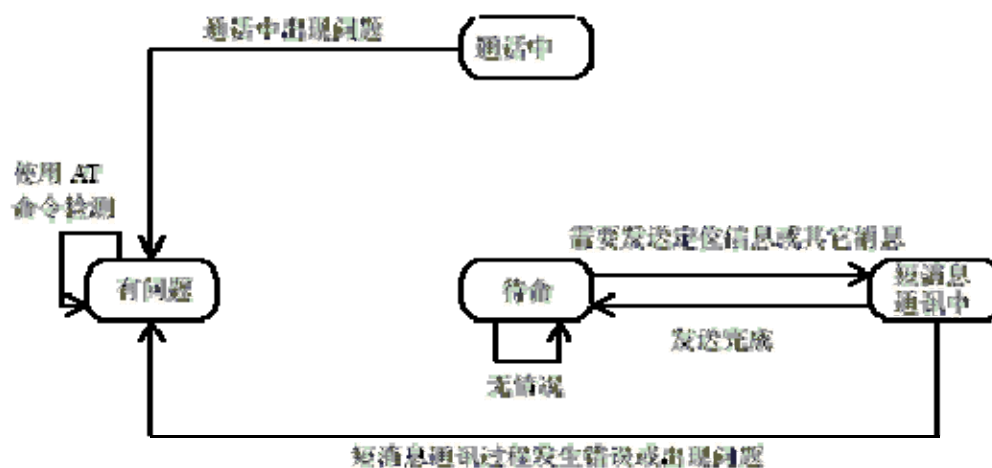


图 1-2 无线电通讯模块状态图

图 1-3 为车载终端用户在遇到特殊情况下通过车载电话或按键与调度中心保持通讯的处理过程顺序图。

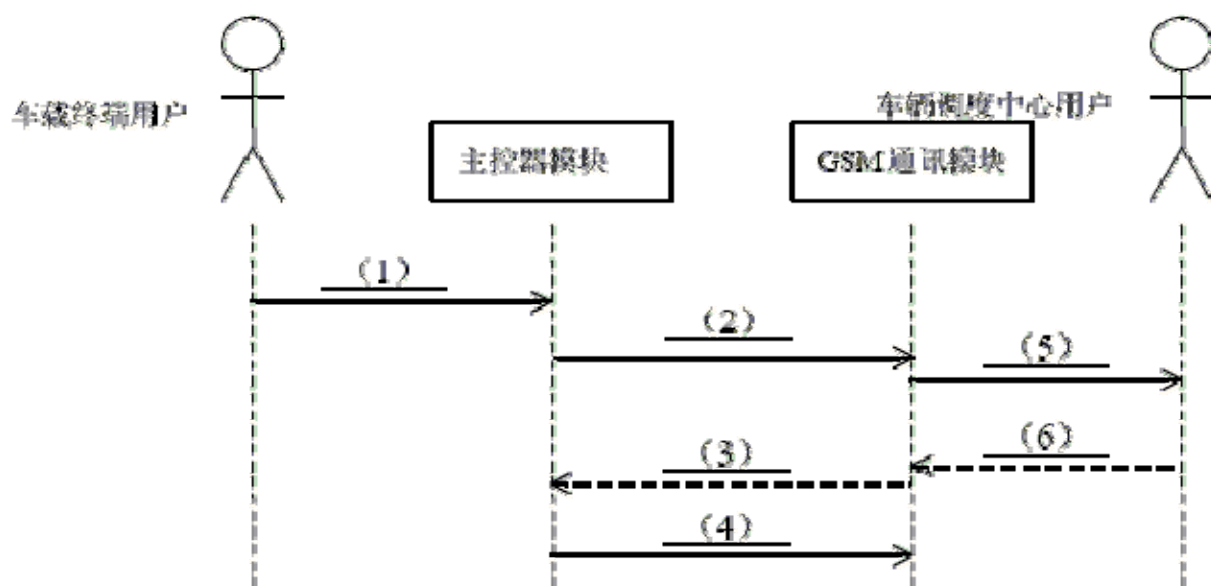


图 1-3 处理过程顺序图

【问题 1】 (4 分)

图 1-2 的状态图不完全, 请将其补充完整, 并将下面给出的转换关系填入图 1-2 的适当位置。

转换关系：① 用户需要语言通话 ② 通话完成
 ③ 重新连接网络 ④ 未找到网络或网络出错

【问题 2】 (6 分)

按照你对 GPS 终端系统的理解,将下列 A~F 标记的处理过程填入到图 1-3 中的 (1)~(6)处,并选择正确的处理顺序。

A. 语音对话/按下按钮 B. 发送信息到通讯模块 C. 监听命令
D. 接收成功要求监听信息 E. 要求监听 F. 通过 GSM
发送信息

处理顺序选项: ① ABDFCE ② ACDBFE ③ ABCDEF ④ ACBDEF

【问题 3】 (5 分)

画出 UML 模型中常用关系符的图示：依赖、泛化、关联、聚合和实现

试题二 (15 分)

阅读以下说明和 X86 汇编语言代码，将应填入 (n) 处的字句写在答题纸的对应栏内。

【说明】

设有数组 X 和 Y。X 数组中有 X1、X2、……、X10 十个数，Y 中有 Y1、Y2、……、Y10 十个数。编制程序计算：

$$Z_1 = X_1 + Y_1 \quad Z_2 = X_2 + Y_2 \quad Z_3 = X_3 - Y_3 \quad Z_4 = X_4 - Y_4$$
$$Z_5 = X_5 - Y_5 \quad Z_6 = X_6 + Y_6 \quad Z_7 = X_7 - Y_7 \quad Z_8 = X_8 - Y_8$$
$$Z_9 = X_9 + Y_9 \quad Z_{10} = X_{10} + Y_{10}$$

设置一个 10 位的标志尺: LOGIC RULE, 若它的每一位决定了当前按顺序该

做加法（该位为 0）或减法（该位为 1）操作，则由上列等式可知 LOGIC_RULE 中的低 10 位应是（1）。

【汇编程序代码】

```

    dataarea    segment                                ;define data segment
    x           dw      x1, x2, x3, x4, x5, x6, x7, x8, x9, x10
    y           dw      y1, y2, y3, y4, y5, y6, y7, y8, y8, y10
    z           dw      z1, z2, z3, z4, z5, z6, z7, z8, z9, z10
    logic_rule  dw （2）                                ;LOGIC_RULE
initialize
    dataarea    ends
    program     segment                                ;define code segment
    main        proc far
                assume cs:program, ds:dataarea

                start:                                ;starting execution
address        （3）                                ;save old data
segment
                sub     ax, ax                        ;put zero in AX
                push    ax                            ;save it on stack
                ;set DS register to current data segment
                mov     ax, dataarea
                mov     ds, ax
                ;MAIN PART OF PROGRAM GOES HERE
                mov     bx, 0
                mov     cx, 10
                mov     dx, logic_rule

next:
                mov     ax, x[bx]
                shr     dx, 1
                jc      subtract
                （4）
                jmp     short result

subtract:
                sub     ax, y[bx]

result:
                （5）                                ;result=>Zn
                add     bx, 2
                loop    next
                ret                                     ;return to DOS
    main        endp                                ;end of main part of
program
    program     ends                                ;end of code segment
                end     start                        ;end of assembly

```

附：指令及寄存器说明：

MOV	传送字或字节
PUSH	把字压入堆栈
HR	逻辑右移
JC	有进位时转移
JMP	无条件转移指令
SUB	减法
ADD	加法
LOOP	CX 不为零时循环
JNE/JNZ	不等于时转移
CMP	比较

段寄存器 2 个：

CS:	代码段寄存器
DS:	数据段寄存器

数据寄存器 4 个：

AX:	累加器
BX:	基址寄存器
CX:	计数寄存器
DX:	数据寄存器

试题三(15 分)

阅读以下说明和 C 语言代码，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

有两个任务（编号分别为 0 和 1）需要访问同一个共享资源，为了解决竞争条件（race condition）的问题，我们需要实现一种互斥机制，使得在任何时刻只能有一个任务访问该共享资源。代码一给出了一种实现方法。

【代码一】

```
1:  int flag[2]; /* flag 数组，初始化为 FALSE */
2:  Enter_Critical_Section(int my_task_id, int
other_task_id)
3:  {
4:  while (flag[other_task_id] == TRUE); /* ?空循环语
句 */
5:  flag[my_task_id] = TRUE;
6:  }
7:  Exit_Critical_Section(int my_task_id,
int ?other_task_id)
8:  {
9:  flag[my_task_id] = FALSE;
10: }
```

当一个任务要访问临界资源时，就可以调用代码一给出的这两个函数。代码二给出了任务 0 的一个例子。

【代码二】

```
Enter_Critical_Section(0, 1);
.... 使用这个资源
.....Exit_Critical_Section(0,1);
..... 做其他的事情 .....
```

【问题 1】（2 分）

什么是临界资源（critical ?resource）？请用 100 字以内文字简要说明。

【问题 2】（6 分）

代码一所示的方法能够实现共享资源的互斥访问吗？请用 300 字以内文字举例说明。

【问题 3】（4 分）

代码一采用了一种繁忙等待(busy waiting)的策略，这种策略的缺点是什么？请用 100 字以内的文字简要说明。

【问题 4】（3 分）

如果把 Enter_Critical_Section() 函数中的两条语句互换一下位置，会出现什么情况？

试题四(15 分)

阅读以下说明和 C 语言代码，回答问题 1 至问题 5，将解答填入答题纸的对应栏内。

【说明】

在实模式存储管理方案下，嵌入式系统的内存地址空间的布局一般可以分为五个段：

代码段（text）、数据段（data）、bss 段（bss）、堆（heap）和栈（stack）。

图 4-1 为一段例程。

```
1: #include <malloc.h>
2: unsigned char gvCh;
3: unsigned short gvShort;
4: unsigned int gvInt = 0x12345678;
5: unsigned long gvLong = 0x87654321;
6: void main(void)
7: {
8:   unsigned char array[10],*p;
9:   p = malloc(10*sizeof(char));
10:   while (1);
11: }
```

图 4-1

【问题 1】（2 分）

代码段、数据段和 bss 段的大小是在什么时候确定的？

【问题 2】（3 分）

常用的堆空间的管理方法主要有哪两种？

【问题 3】（2 分）

当图 4-1 所示的程序经过编译链接并装入内存时，与 while 语句相对应的可执行代码存放在哪一个段？

【问题 4】（6 分）

在图 4-1 所示的程序中，变量 gvCh、gvShort、gvInt、gvLong、array、p 分别存放在哪一个段？

【问题 5】（2 分）

malloc 函数申请的空间位于哪一个段？

试题五 (15 分)

阅读以下关于嵌入式系统中异步串口的叙述，回答问题 1 至问题 4。

【说明】

异步串口是嵌入式处理器上最常用资源之一。一般而言，异步传输的数据以帧的方式传输。每一帧有效数据前有一个起始位，帧结束于一个或多个停止位。异步串口的数据由起始位和停止位分割成数据帧。常用的异步串口数据帧格式如图 5-1 所示。

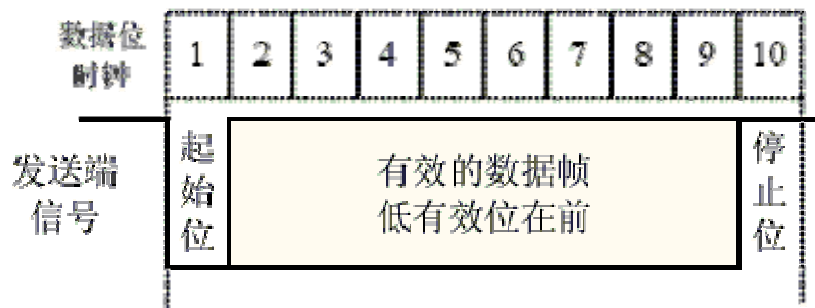


图 5-1 异步串口数据帧格式

RS-232、RS-422 和 RS-485 都是常用的异步串口标准，它们的时序完全一样，只是在电气特性上有所区别，它们之间通过通用异步收发器（UART）可实现转换。UART 控制器可以集成到芯片中或者通过嵌入式处理器总线连接，所以，通常从 UART 发出的异步串口时序的逻辑电平都是处理器 I/O 电压标准（比如：TTL、LVTTTL 等标准）。若要求符合 RS-232、RS-422 或者 RS-485 的电气特性，则需要接口电路做转换。使用 MAX3232 芯片实现的串口电平转换的电路原理图如图 5-2 所示。

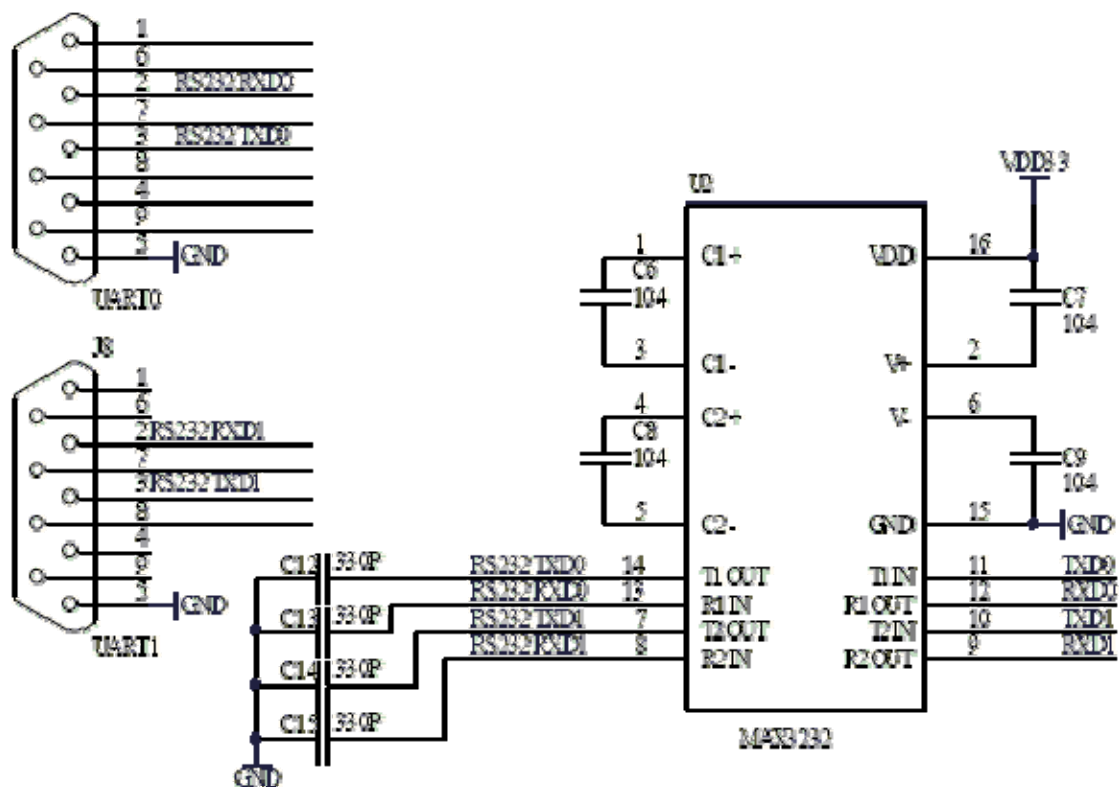


图 5-2 使用 MAX3232 芯片实现的串口电平转换的电路原理图

【问题 1】（3 分）

请说明异步传输和同步传输的不同之处。

【问题 2】（4 分）

根据图 5-1，请用 300 字以内文字简要描述异步串口的数据传输过程。

【问题 3】（4 分）

如果系统设计采用串行数据传输最高波特率为 115200bps、16 倍分频，则时钟至少为多少赫兹？

【问题 4】（4 分）

图 5-2 所示的电路原理图设计实现了几路串口？每个串口为多少条连接线？