

2006 年下半年嵌入式系统设计师答案

上午答案

(1)	C	(16)	D	(31)	C	(46)	A	(61)	D
(2)	A	(17)	B	(32)	B	(47)	B	(62)	D
(3)	C	(18)	D	(33)	C	(48)	B	(63)	A
(4)	B	(19)	C	(34)	B	(49)	A	(64)	B
(5)	C	(20)	B	(35)	D	(50)	C	(65)	A
(6)	B	(21)	D	(36)	D	(51)	B	(66)	B
(7)	D	(22)	A	(37)	B	(52)	A	(67)	D
(8)	A	(23)	C	(38)	D	(53)	D	(68)	C
(9)	B	(24)	C	(39)	B	(54)	D	(69)	A
(10)	C	(25)	D	(40)	B	(55)	C	(70)	C
(11)	B	(26)	B	(41)	D	(56)	B	(71)	C
(12)	C	(27)	C	(42)	C	(57)	A	(72)	D
(13)	B	(28)	B	(43)	A	(58)	C	(73)	B
(14)	A	(29)	B	(44)	D	(59)	A	(74)	A
(15)	A	(30)	B	(45)	B	(60)	A	(75)	C

下午答案

试题一

[问题 1]

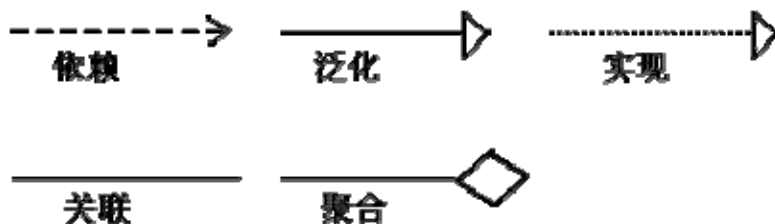


[问题 2]

- (1) A, 或语音对话/按下按钮
- (2) B, 或发送信息到通讯模块
- (3) E, 或要求监听
- (4) C, 或监听命令
- (5) F, 或通过 GSM 发送信息
- (6) D, 或接收成功要求监听信息

正确的处理顺序为: ③, 或 ABCDEF

[问题 3]



试题二

- (1) 00 1101 1100
- (2) 00dch
- (3) push ds
- (4) add ax, y[bx]
- (5) mov z[bx], ax

试题三

[问题 1]

在多道程序系统中, 各进程可以共享各类资源, 但有些资源一次只能供一个进程使用, 需要互斥访问的资源称为临界资源, 如打印机、共享变量、表格等。

[问题 2]

代码一所示的方法不能实现资源互斥访问。

例如, 考虑如下的情形:

- (a) 初始化的时候, flag 数组的两个元素值均为 FALSE
- (b) 任务 0 先执行, 在执行 while 循环语句时, 由于 flag[1]=FALSE, 所以顺利结束, 不会被卡住。假设这个时候来了一个时钟中断, 打断它的运行;
- (c) 任务 1 去执行, 在执行 while 循环语句时, 由于 flag[0]=FALSE, 所以顺利结束, 不会被卡住, 然后就进入了临界区;
- (d) 后来当任务 0 再执行时, 也进入了临界区, 这样就同时有两个任务在临界区

[问题 3]

简要说明: 繁忙等待的基本思路是: 当一个任务要进入临界区, 首先需要检查是否允许它进入, 若允许, 则直接进入; 否则, 则循环等到, 使用一个循环语句不断执行测试指令, 从而浪费了大量的 CPU 资源。

[问题 4]

可能会出现死锁

试题四

[问题 1]

在编译时确定。

[问题 2]

固定分区、可变分区。

[问题 3]

代码段。

[问题 4]

全局变量 gvCH 和 gvShort 由于没有设置初始值, 所以放在 bss 段当中。

全局变量 gvInt 和 gvLong 有初始值, 所以放在 data 段当中。

指针 p 和数组 array, 它们都是 main 函数的局部变量, 所以存放在栈当中。

[问题 5]

堆空间。

试题五

[问题 1]

异步通信系统的数据传输, 其接收器时钟与发送器时钟是不同步的。

[问题 2]

一帧数据是从最低有效位开始传输的。

传输线在逻辑 1 时处于空闲状态。当每帧数据开始传输时, 先发送一个起始位。起始位是从逻辑 1 到 0 的迁移, 这个变化表明一帧数据开始传输, 接收端则在开始检测到起始位的时候, 按照收发两端事先约定好的通讯速度, 检测后面的数据位, 从而组成一帧数据。

在传输的最后, 利用 1 个停止位(逻辑 1)使传输线回到空闲状态。然后发送方才可以发送下一个字节的数据。

[问题 3]

如果设计系统的最高波特率标准为 115200bps, 则时钟至少为
 $115200 \times 16 = 1.8432\text{MHz}$

[问题 4]

实现了两路的串口, 每个串口为 3 条连接线。