

2021 天勤计算机考研八套模拟卷 · 卷七

组成原理篇选择题答案解析

1. A。
由于 x 的符号位为 1，可知 x 为负数。又因为 $x > -2^{n-1}$ ，可以得到 x 的绝对值必须小于 2^{n-1} ，所以 x_{n-1} 必须为 0。
2. B。
具体请参看下表。

原码、补码的乘法与除法		
运算种类	符号位处理	大致规则
原码乘法	符号位异或	乘数 Y_i 为 1 时，被乘数绝对值与原部分积相加后，右移一位； 乘数 Y_i 为 0 时，直接右移一位
补码乘法	不单独处理	(1) 被乘数 x 符号任意，乘数 y 为正：同原码一位乘 (2) 被乘数 x 符号任意，乘数 y 为负：先将 $[y]_{补}$ 去掉符号位，进行原码一位乘操作，得到的结果进行 $[-x]_{补}$ 校正 (3) Booth 算法：乘数连同符号位一同加入运算，并且增加一位附加位 y_{n+1} ，其初始值为 0。每一次计算时，需要查看 $y_i y_{i+1}$ ：00 和 11 时，部分积右移一位；01 时，部分积加 $[x]_{补}$ ，再右移一位；10 时，部分积加 $[-x]_{补}$ ，再右移一位。（注意：Booth 算法在最后一步是不移位的，即如果出现 00 和 11 则计算结束；若出现 10 或 01，则部分积加相应值后结束计算）
原码除法	符号位异或	(1) 恢复余数法：首先将被除数绝对值 $[-y^*]_{补}$ ，若余数为正，上商“1”，左移一位，然后继续进行该操作；若余数为负，上商“0”，恢复余数 $+[y^*]_{补}$ ，如此重复 (2) 加减交替法：首先将被除数绝对值 $[-y^*]_{补}$ ，若余数为正，上商“1”，左移一位， $+[y^*]_{补}$ ；若余数为负，上商“0”，左移一位， $+[y^*]_{补}$
补码除法	不单独处理	首先判断 $[x]$ 和 $[y]$ 是否同号，若同号，则 $[-y]_{补}$ ；若异号，则 $[y]_{补}$ 。然后，查看余数 $[R]_{补}$ 和 $[y]_{补}$ 是否同号，若同号，则上商“1”，左移一位，然后 $[-y]_{补}$ ；若异号，左移一位，然后 $[y]_{补}$ ，如此重复

- 注：1. 只要是原码运算，符号位一定是单独处理，不参与运算。
2. 此表只是列出大体的运算规则，具体的操作还需要读者进行一些针对的练习，虽然作为大题考到的概率几乎为 0，但是也应该从练习中多总结一些规律，以应付选择题。

- 3.C。
因为主存按字节编址，每块 32B，故第 3000 号单元（从 0 开始编制）所在的块号为 $3000/32=93$ ；又因为 Cache 采用四路组相连，一共 64 行（可看成 64 块），一共有 $64/4=16$ 组，于是按照主存块号对应 Cache 组号，映射后第 93 块在 Cache 中的组号为 $93\%16=13$ 。答案选 C。
提示：求第多少号单元属于第几主存块的时候，很多同学可能有这样的错觉，只要不是整除，那么就一定在下一个主存块，其实这是不对的，**因为块的编号是从 0 开始的**，原则上从次序上来说，它确实是第 94 块，但计算机起始计数一般都是从 0 开始的，这样它的序号就变成了 93，Cache 也是如此，因此当结果求得是 13 的时候，它确实是在次序上来说处于第 14 组，但是编号确实第 13 组；另外，题目不加说明我们默认主存块大小等于 Cache 块大小，Cache 一行等于一块。

- 4.D。
这是一个部分译码的片选信号（因为高 8 位地址中有两位没有参与译码），根据译码器电路，译码输出的逻辑表达式应为

$$\overline{CS} = A_{19}(A_{18}+A_{17})A_{15}A_{13}A_{12}$$

- 注意：** ≥ 1 表示只要有一个为 1 即可，所以形成 $A_{17}+A_{18}$ 。而译码器中间有一个 $\&$ ，所以 A_{19} 、 $A_{17}+A_{18}$ 、 A_{15} 、 A_{13} 、 A_{12} 都必须为 1。换句话说， A_{19} 、 A_{15} 、 A_{13} 、 A_{12} 必须为 1，而 A_{17} 、 A_{18} 必须至少有 1 个为 1。
由于 D 选项的 A_{12} 为 0，所以不属于此译码空间。

5. D。
相同的题型已经在第三套卷子的第 17 题已经讲过；操作码不固定，有 m 条双操作数指令，所以前 8 位还剩下

256-m 条。有 n 条无操作数指令, 所以还剩下的空间只有 $[(2^8-m) \times 2^{12}-n]$, 即可设计出 $[(2^8-m) \times 2^{12}-n]/2^6$ 条, 结果取整; 当然这里也可以按照第三套当中讲的第二种方法, 这里不做过多赘述。

6. B。

I: I1 指令运算结果应先写入 R1, 然后在指令 I2 中读出 R1 的内容。由于 I2 指令进入流水线, 使得 I2 指令在 I1 指令写入 R1 前就读出 R1 的内容, 发生“写后读相关”。

II: I1 指令应先读出 R2 的内容并存入存储单元 M 中, 然后 I2 指令将运算结果写入 R2 中。但由于 I2 指令进入流水线, 使得 I2 指令在 I1 指令读出 R2 之前就写入 R2, 发生“读后写相关”。

III: I2 指令应该在 I1 指令写入 R3 之后, 再写入 R3。现由于 I2 指令进入流水线, 如果 I2 指令减法运算在 I1 指令的乘法运算之前完成, 使得 I2 指令在 I1 指令写入 R3 之前就写入 R3, 导致 R3 内容错误, 发生“写后写相关”。

7. B。

修改之后的优先级是 3、1、2、4, 表示 3 号的优先级最高, 它可以抢占任何级别的中断处理, 故 3 的屏蔽字是 1111, 1 号其次, 表示除了 3 号我不能抢占它, 其余的我都可以抢占, 故 1 的屏蔽字为 1101, 依次类推, 2 号中断屏蔽字为 0101, 4 号中断屏蔽字为 0001。

记忆: 谁比我厉害, 我的屏蔽字那一位就是 0, 我比谁厉害, 我的屏蔽字那一位就是 1。

8. B。

设计微指令结构时, 所追求的目标如下:

- ① 微指令结构要有利于缩短微指令字长度。
- ② 有利于减小控制存储器的容量。
- ③ 有利于提高微程序的执行速度。
- ④ 有利于微指令的修改。
- ⑤ 有利于微程序设计的灵活性。

9. C。

A: CPU 通过总线的类型来识别信息是地址还是数据, 故 A 选项错误。

B: 间接寻址第一次访问内存所得到的信息是操作数的有效地址, 该地址通过**数据线**传送至 CPU, 而不是地址线, 故 B 选项错误。

C: 在单总线结构中, CPU、主存和 I/O 设备 (通过 I/O 接口) 都挂在一组总线上, 若 I/O 设备和主存统一编址, 则可以很方便地使用访存指令访问 I/O 设备, 故 C 选项正确。

D: 异步总线即采用异步通信方式的总线。在异步方式下, 没有公共的时钟, 完全依靠传送双方相互制约的“握手”信号来实现定时控制, 故 D 选项错误。

10. C。

不互锁方式的请求信号和回答信号没有相互的制约关系, 主设备在发出请求信号后, 不必等待回答信号的到来, 便自己撤销了请求信号好, 所以速度最快。