

2021 天勤计算机考研八套模拟卷 · 卷六

组成原理篇选择题答案解析

1. C。

I: 对于原码表示的基值为 4 的小数, 规格化的形式是小数点后 2 位不全为 0, 故 I 错误。

最笨的解题思路: 基数 $r=4$, 由于 $1/r \leq |M| \leq 1$, 即尾数的十进制绝对值在 $0.25 \sim 1$ 之间。而 $(0.000\ 010)_2 = 0.03125$, 故不是规格化数。

II: 浮点数的溢出并不是由尾数来判断的, 而是规格化后阶码超出所能表示的范围时, 才表示溢出, 故 II 错误。

III: 在浮点数的运算过程中, 尾数如果出现 $01.XXX \dots X$ 和 $10.XXX \dots X$, 则需要右规, 并且只需进行一次右规尾数就会变成规格化数, 但是左规操作可能不止一次, 故 III 正确。

2. D。

I: 在原码一位乘算法过程中, 符号位是不参加运算的, 结果的符号位是被乘数的符号位和乘数的符号位异或的结果, 故 I 错误。

II: 在原码一位乘算法过程中, 由于参与操作的数是真值的绝对值, 所以没有正负可言, 故在原码一位乘法中运算过程中所有的移位均是逻辑移位操作, 即在高位添加 0, 故 II 错误。

III: 由于在部分积相加中, 可能导致两个小数相加大于 1, 所以部分积至少需要使用 $n+1$ 位寄存器, 故 III 错误。

3. A。

很多不了解 DRAM 引脚结构的同学很可能会得出 $24+8=32$ 的结果, 其实这是不正确的, 在《高分笔记》当中讲过半导体存储芯片的译码驱动方式, 其中介绍了重合法, 将存储单元分成行和列, 然后分别通过行地址线和列地址线来确定行列地址从而确定一个单元, 这里 DRAM 采用引脚复用, 将行地址线和列地址线合用作一组, 只不过在译码时, 需要发送两次地址信号 (相当于一次行地址, 一次列地址), 从而减少了 DRAM 的引脚总数, 便于设计 DRAM; 因此这里地址空间是 16M, 需要 24 个地址位来标识, 分为两次发送, 则地址引脚数为 12, 故地址引脚和数据引脚总数为 $12+8=20$ 。

【总结】 DRAM 芯片采用引脚复用, 且行列地址位数一致。这种题在 14 年统考真题当中已经出现过。

4. A。

不妨设地址线和数据线的数目分别为 x 和 y 。

只需要满足 $2^x \times y = 64K \times 2$, 所以就有如下方案:

当 $y=1$ 时, $x=17$;

当 $y=2$ 时, $x=16$;

当 $y=4$ 时, $x=15$;

当 $y=8$ 时, $x=14$;

后面的就不要计算了, 肯定比前面的引脚数目多。从以上分析可以看出, 当数据线分别为 1 或 2 时, 地址线和数据线引脚的数目之和为 18, 达到最小, 并且有两种解答。

5.B。

首先可以直接排出 C、D 选项, 因为无论偏移量是多少位, 由于偏移量是采用补码表示的, 根据补码的特性, 它比源码表示的数多一位, 而且多出来的就是补码的最小值。因此偏移量的最小值一定是一个偶数。操作码占 8 位, 两个操作数具有两种不同的寻址方式, 则需要 2 位寻址特征位, 另外一共有 30 个寄存器, 故需要 5 位来标识选择哪个寄存器, 所以偏移量的位数 $= 32 - 8 - 2 - 5 - 5 = 12$, 而 12 位的带符号的补码所能表示的数的范围为 $-2048 \sim 2047$ 。

【提示】 在考场上有时候即使我们不能一步就算出结果, 或者题目复杂的时候, 可以抓住问题的一些细节来排除某些选项, 这对我们分析余下的选项也是很有帮助的。

6. C。

相对寻址本身就是相对于本指令地址进行上下浮动, 所以相对寻址的区间范围和本指令的地址密切相关, 其他 3 个选项都与本指令的地址无关。

7. D。

指令的各个子功能在不同的部件中是并行执行的, 因此执行这条指令的时间一定是各个子功能中所花的最长时间, 当前最长时间为 80ns, 当合并 E 和 F 这两个功能部件之后, 合并子功能执行时间为 50ns, 因此最长的时间还是 80ns, 再加上 20ns 的寄存器延迟, 所以五段流水线的时钟周期至少是 100ns。

8. D。

本题问的是微程序中首条微指令的地址, 稍不注意就可能误选 B, 微程序是用来解释指令的, 通过指令操作码的内容来区别指令, 然后根据指令操作码映射找到对应解释这个指令的微程序段。因此首条微指令的地址是由指令操作码映射而来的。

9. B。

在流水线处理器中处理数据相关问题有两种方法: 一种是暂停相关指令的执行, 即暂停流水线, 直到能够正确读出寄存器操作数为止; 另一种是采用旁路电路技术, 即采用专门的数据通路, 直接把结果送到 ALU 的输入端, 也就是把内部数据前推, 即不必等待某条指令的执行结果写回到寄存器后, 再从寄存器取出结果, 而是直接将执行结果通过专用通路送至需要该结果的地方。

10. D。

当每次计数从 $\lfloor n/2 \rfloor$ 开始时, 所有设备被分为两部分, 设备号为 $\lfloor n/2 \rfloor$ 到 n 的设备优先级高于设备号为 0 到 $\lfloor n/2 \rfloor - 1$ 的设备; 且在这两部分内, 却是设备小的优先级高, 故 A、B、C 选项都是错误的。

11. D。

通道的工作过程如下:

(1) 用户程序中使用访管指令进入操作系统的管理程序, 由 CPU 通过管理程序组织一个通道程序, 并使用 I/O 指令启动通道 (此后 CPU 就可以并行运行应用程序了)。

(2) 通道并行执行 CPU 为它组织的通道程序 (通道程序在主存中), 完成指定的数据输入输出工作。

(3) 通道程序结束后向 CPU 发出中断请求。CPU 响应这个中断请求后, 第二次调用管理程序对输入输出中断请求进行处理。

这样, 每完成一次输入输出工作, CPU 只需要两次调用管理程序, 大大减少了对用户程序的打扰。

补充: 在采用通道结构的系统中, 也需要使用 I/O 指令, 但这种 I/O 指令比较简单, 它并不直接控制具体 I/O 操作, 只是负责通道的启动和停止、查询通道或设备的状态, 从而控制通道去完成 I/O 操作。