

第一次 期末复习

1. 下列部件的位数一定与机器字长相同的是 ()

I. ALU II. 指令寄存器 III. 通用寄存器 IV. 浮点寄存器

- A. I、II
- B. I、III
- C. II、III
- D. II、III、IV

查看答案和解析

答案：B

解析：

首先，ALU 是用来进行算术运算的主要部件，一定与机器字长相同，通用寄存器则可以用来保存中间计算结果，比如乘法器，这也需要与机器字长相同等。

拓展：指令寄存器（Instruction Register，简称IR）是计算机中的一种特殊寄存器，用于存储当前正在执行的指令。所以其位数通常与计算机体系结构和指令集架构有关，与指令长度，寻址方式等密切相关。

浮点寄存器（Floating Point Register）是计算机中用来存储浮点数（floating-point numbers）的寄存器。通常需要更多位数进行拓展以提高计算精度。

2. 下列关于冯·诺依曼结构计算机基本思想的叙述中，错误的是 (C)

C 错误，因为数据不一定都直接在指令中给出。它可以存储在存储器的某个地址中，然后在执行指令时按地址访问这个数据。类似于一种间接访问的形式。

- A. 程序的功能都通过中央处理器执行指令实现
- B. 指令和数据都用二进制表示，形式上无差别
- C. 指令按地址访问，数据都在指令中直接给出
- D. 程序执行前，指令和数据需预先存放在存储器中

3. $93PFLOPS = (D)$

D正确，首先需要知道 P 是多大 注意这里的 P 是用来衡量计算速度的而不是文件大小
在衡量大小时 $1K = 2^{10}$ $1M = 2^{20}$ $1G = 2^{30}$ $1T = 2^{40}$ $1P = 2^{50}$

在衡量速度时 $1K = 10^3$ $1M = 10^6$ $1G = 10^9$ $1T = 10^{12}$ $1P = 10^{15}$

此题中即 $93 \times 10^{15} = 9.3 \times 10^{16}$ 而 $10^8 = 1$ 亿

所以选择D 9.3亿亿

- A. 9.3×10^{13}
- B. 9.3×10^{15}
- C. 9.3千万亿
- D. 9.3亿亿

4. 下列关于冯·诺依曼计算机中指令和数据存放位置的叙述，正确的是 (D)

D正确，一开始计算机中指令和数据存放在外存中，启动后被装入内存

- A. 指令存放在内存中，数据存放在外存中
- B. 指令和数据任何时候都存放在内存中
- C. 指令和数据任何时候都存放在外存中
- D. 程序被启动前指令和数据都存放在外存中，而启动后指令和数据都被装入内存中

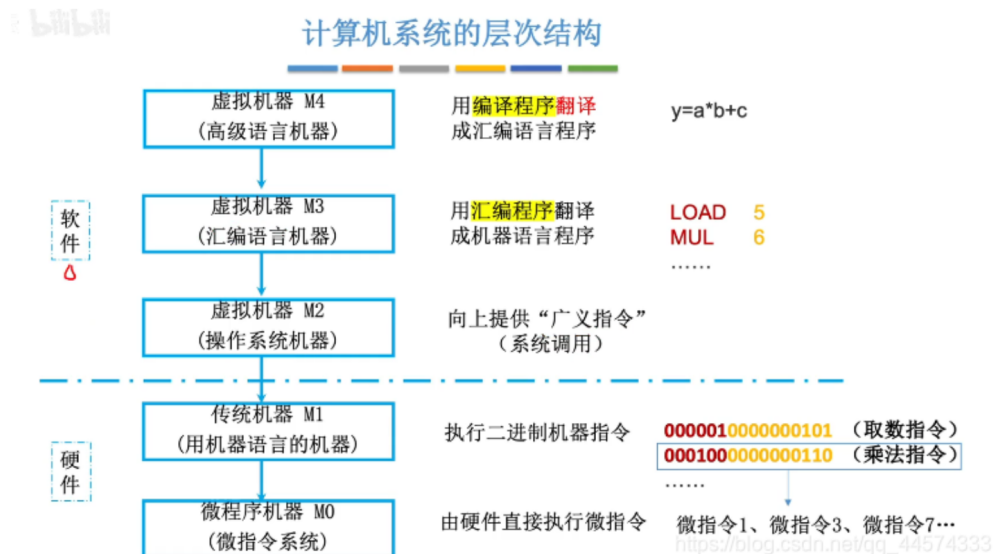
5. 一般情况下, "裸机"是指 (C)

C正确，裸机是指没有安装任何软件的计算机。计算机由硬件系统和软件系统所组成，没有安装任何软件的计算机称为裸机

- A. 单片机
- B. 没有使用过的计算机
- C. 没有安装任何软件的计算机
- D. 只安装操作系统的计算机

6. 计算机系统的层次结构，下列五个级别的机器由下到上的顺序是 (B)

B正确，计算机系统的层次结构见下图。



I 机器语言机器 II 汇编语言机器 III 高级语言机器 IV 微程序控制机器 V 操作系统机器

- A. I \rightarrow II \rightarrow III \rightarrow IV \rightarrow V

- B. $IV \rightarrow I \rightarrow V \rightarrow II \rightarrow III$
- C. $III \rightarrow II \rightarrow V \rightarrow I \rightarrow IV$
- D. $V \rightarrow IV \rightarrow III \rightarrow II \rightarrow I$

7. 下列关于系统软件叙述中，正确的是 (A)

A正确，系统软件和应用软件共同组成了计算机软件系统。直接支持用户使用计算机硬件，是计算机系统中最接近硬件的一层通用软件，一般与具体应用领域无关。

- A. 系统软件与具体应用领域无关
- B. 系统软件与具体硬件逻辑功能无关
- C. 系统软件是在应用软件基础上开发的
- D. 系统软件并不具体提供人机界面

8. 对于汇编语言程序员来说，以下部件不透明的是 (C)

C正确，首先需要知道透明的意思，在计算机的术语中透明的意思是看不见，而不透明的意思是看得见。透明部分是指那些不属于自己管的那部分，所以看不见。而不透明则需要自己了解和管理。汇编语言程序员需要通过汇编语言来直接读写通用寄存器中的值以进行运算，可以想象以下汇编语言的语法，可以操作通用寄存器中的值来进行运算。同时需要操作中断字寄存器来修改中断的优先级。

拓展: 关于计算机中透明与不透明的寄存器

所有用户可见（应用程序员等于用户）：

PC(指令计数器), PSW(标志寄存器), 通用寄存器, ACC(累加器)

对汇编程序可见（用户可见的汇编程序员也可见）：

中断字寄存器（可以修改中断的优先级）

基址寄存器（基址寻址）

变址寄存器（变址寻址，如数组的访问需要）

条件码寄存器

通用寄存器

对所有用户不可见：

MAR(存储器地址寄存器)

MDR(存储器数据寄存器)

IR(指令寄存器)（指令寄存器是从主存中取出来存放指令的寄存器，无法编辑，自然也不可
见）

uIR(微指令寄存器, uMAR, uMDR)

Cache(高速缓存)

I 指令缓冲器 II 移位器 III 通用寄存器 IV 中断字寄存器 V 乘法器 VI 先行进位链

- A. I、II、和 III
- B. IV、V 和 VI

- C. III 和 IV
- D. I、II、V、VI

9. 下列说法中错误的是 (D)

D错误，主频是只CPU时钟周期的倒数，而不是机器周期的倒数。

- A. 主频为 $1GHz$ 的 CPU 运算性能可能强于主频为 $2GHz$ 的 CPU
- B. 衡量用于科学计算的 CPU 性能的主要指标为 MFLOPS
- C. 字长指计算机能直接处理的二进制信息的位数
- D. 主频是指 CPU 机器周期的倒数

10. 计算机的机器字长为 32 位，下列说法中正确的是 (A)

A正确，CPU一次能够处理的数据长度称为机器字长。通常机器字长等于通用寄存器的长度。

通用寄存器是指一种多用途寄存器，可用于保存源操作数和计算得到的中间结果，也可以作为变址寄存器，还可以存放基地址和其他信息等。

而数据总线的宽度取决于计算机系统的数据传输需求。它通常指定为每个时钟周期可以传输的数据位数的总数。数据总线的宽度越大，系统在单位时间内能够传输的数据量就越大。与数据传输的数据量有关

拓展: 三大总线 数据总线，地址总线，控制总线

数据总线

- (1) 是CPU与内存或其他器件之间的数据传送的通道。
- (2) 数据总线的宽度决定了CPU和外界的数据传送速度。
- (3) 每条传输线一次只能传输1位二进制数据。eg: 8根数据线一次可传送一个8位二进制数据(即一个字节)。
- (4) 数据总线是数据线数量之和。

地址总线

- (1) CPU是通过地址总线来指定存储单元的。
- (2) 地址总线决定了cpu所能访问的最大内存空间的大小。eg: 10根地址线能访问的最大的内存为1024位二进制数据 (1024个内存单元)
- (3) 地址总线是地址线数量之和。
- (4) 地址总线是计算机系统中用于传输地址信息的总线，它与存储器和I/O设备进行通信，确定要读取或写入的位置。通过地址总线传输的地址码，计算机系统可以准确定位存储器或I/O设备的位置，实现数据的读写操作。

控制总线

- (1) CPU通过控制总线对外部器件进行控制。
- (2) 控制总线的宽度决定了CPU对外部器件的控制能力。
- (3) 控制总线是控制线数量之和。

最后总结如下：

地址总线的宽度决定CPU的寻址能力；

数据总线的宽度决定CPU与其他元器件一次最大传送的数据量；

控制总线决定CPU对其他元器件的控制能力。

- A. 通用寄存器一般为 32 位
- B. 数据总线宽度为 32 位
- C. 支持 64 位操作系统
- D. 以上说法均不正确

11. 若一台计算机的机器字长为 4 字节，则表明该机器 (C)

C 正确，一个字节(1B) = 8b，4 个字节 即 32 位 故选择 C。

一般来说 32 位运算的最大结果位 $2^{31} - 1$ 因为使用补码存储的数据，最高位始终为符号位。故少一位，最大能表示的数相当于 31 位二进制数 $2^{31} - 1$ 。

- A. 能处理的数值最大为 4 位十进制数
- B. 能处理的数值最多位 4 位二进制数组成
- C. 在 CPU 中能够作为一个整体处理 32 位二进制代码
- D. 在 CPU 中运算的结果最大位 2^{32}

12. MIPS (每秒百万次指令数) 和 MFLOPS (每秒百万次浮点运算数) 是衡量 CPU 性能的两个指标其中

MIPS 适合衡量标量处理器的性能，MFLOPS 适合衡量向量处理器的性能

了解即可

13. 运算发生溢出的根本原因是 (A)

A 正确，无论采用何种机器数，只要运算结果大于数值设备所能表示数的范围，就会产生溢出。

- A. 数据的位数有限
- B. 运算中符号位的进位丢弃
- C. 运算中将符号位的借位丢弃
- D. 数据运算中的错误

14. 当顶点运算发生溢出时，应 (C)

C 正确，A、B、D 选项都是浮点运算时的操作。

拓展：规格化是指调整浮点数的阶码和尾数，尾数部分用纯小数给出，且小数后第一位的值不为 0。

- A. 向左规格化
- B. 向右规格化
- C. 发出出错信息
- D. 舍入处理

15. 在补码加法运算中,产生溢出的情况是 (C)

C 正确，常用的溢出判断方法主要有三种：采用一个符号位、采用进位位和采用变形补码

采用一个符号位的溢出条件为：结果的符号位与操作数符号位不同。

采用进位位的溢出条件为：结果的符号位和最高数位不同时产生进位。

采用双符号位（变形补码）的溢出条件为：运算结果的两个符号位不同。

可以这样记，不同即溢出

I 两个操作数的符号位相同,运算时采用单符号位,结果的符号位与操作数相同

II 两个操作数的符号位相同,运算时采用单符号位,结果的符号位与操作数不同

III 运算时采用单符号位,结果的符号位和最高数位不同时产生进位

IV.运算时采用单符号位,结果的符号位和最高数位相同时产生进位

V.运算时采用双符号位,运算结果的两个符号位相同

VI.运算时采用双符号位,运算结果的两个符号位不同

- A. I,III,V
- B. II,IV,VI
- C. II,III,VI
- D. I, II,VI

16. 将采用 8 位二进制补码表示的十进制数 -121 拓展成 16 位 二进制补码，结果用十六进制表示为 (B)

B 正确，十进制二进制 -121 的 8 位二进制原码表示为 11111001 拓展成16位则为 1000000001111001

变为补码为 1111111110000111 变为 16 进制为 FF87H。

- A. 0087H
- B. FF87H
- C. 8079H
- D. FFF9H

17. 在串行进位的并行加法器中，影响加法器运算速度的关键因素是 (C)

C正确，四个选择均会对加法器的运算速度产生影响，但进位传递延迟的影响最为关键。因为必须要等待前一位运算完成才能进行下一位的运算。这样效率会受到很大影响。

- A. 门电路的延迟
- B. 元器件速度
- C. 进位传递延迟
- D. 各位加法器的速度的不同

18. 下列叙述中，不正确的是 (A)。

A 不正确

选项A描述的原因是不正确的。串行进位加法器的加法时间长主要是因为它的进位是串行传递的，即每一位的加法结果依赖于前一位的加法和进位结果，因此必须等前一位计算完成后才能进行下一位的计算。这与串行进位加法器的高位电路复杂性无关。

选项B正确，因为在串行进位加法器中，加法的每一位都需要等前一位完成，因此位数越多，加法所需的时间也就越长。

选项C描述的是先行进位加法器的特点之一。先行进位加法器设计复杂，尤其是当位数增加时，为了实现快速进位，其电路设计会更加复杂。

选项D正确描述了串行进位加法器和先行进位加法器加法时间差异的真正原因，即进位的串行传递导致了加法时间的增加。

因此，不正确的叙述是A

- A. 串行进位加法器比先行进位加法器的加法时间长的原因是串行进位加法器高位电路复杂
- B. 串行进位加法器位数越多加法时间越长
- C. 先行进位加法器位数越高位电路越复杂
- D. 串行进位加法器比先行进位加法器的加法时间长的原因是串行进位加法器进位串行传递

19. 计算机中的累加器 (B)

B 正确，累加器又称为累加寄存器，它实质上是寄存器，没有加法器的功能。

- A. 没有加法器功能，也没有寄存器功能
- B. 没有加法器功能，有寄存器功能
- C. 有加法器功能，没有寄存器功能
- D. 有加法器功能，也有寄存器功能

寻址方式期末复习

- 指令的寻址方式：
 - 形式地址 (用 A 表示)
 - 有效地址 (直接访问主存的地址，EA)
- 关系：A 通过寻址方式访问到 EA
- 寻址方式：

$$\text{寻址方式} \left\{ \begin{array}{l} \text{指令寻址} \left\{ \begin{array}{l} \text{顺序寻址}(PC + 1) \\ \text{跳跃寻址(即程序转移指令)} \end{array} \right. \\ \text{数据寻址(寻找操作数的地址)} \end{array} \right.$$

- 寻址方式从快到慢
 1. 立即寻址
 2. 寄存器寻址
 3. 直接寻址 ($EA = A$)
 4. 寄存器间接寻址 ($EA = (Ri)$)
 5. 偏移寻址
 - 变址寻址 ($EA = (R_x) + A$)
 - 相对寻址 ($EA = (PC) + A$)
 - 基址寻址 ($EA = (R_b) + A$)

6. 间接寻址 ($EA = (A)$)

- 表：基本寻址方式比较

| 寻址方式 | 规则 | 优点 | 缺点 |
|---------|----------------|--------|---------------|
| 立即寻址 | 操作数 = A | 无需访问主存 | 操作数取值范围受限 |
| 寄存器寻址 | $EA = R_i$ | 无需访问主存 | 操作空间受限 |
| 直接寻址 | $EA = A$ | 简单 | 寻址空间受限 |
| 寄存器间接寻址 | $EA = (R_i)$ | 寻址空间大 | 比寄存器寻址多访问一次主存 |
| 偏移寻址 | $EA = (R) + A$ | 灵活 | 复杂 |
| 间接寻址 | $EA = (A)$ | 寻址空间大 | 需要多次访问主存 |

- 堆栈：规则为后进先出 (LIFO) (SP 为堆栈指针，指向栈顶)

- 入栈 (SP) - 1 \rightarrow SP, (A) \rightarrow SP
- 出栈 ((SP)) \rightarrow A, (SP) + 1 \rightarrow SP

第二次期末复习

- 采用 16 位定长指令字格式，操作码位数和寻址方式位数固定，指令系统中有 48 条指令支持，直接，间接，立即，相对，4 种寻址方式，单地址指令中直接寻址方式，可寻址范围是 (A)

A 正确，首先这是寻址范围，要满足主存地址不为负数。由题意可知，指令总长为 16 位，有 48 条指令支持，对应的操作码至少需要 6 位，而 4 种寻址方式又需要 2 位，所以形式地址位数为 $16 - 6 - 2 = 8$ ，寻址范围为

$2^8 = 256$ ，即 0 ~ 255。

- A. 0 ~ 255
- B. 0 ~ 1023
- C. -128 ~ 127
- D. -512 ~ 511

- 下列寻址方式种，最适合按下标顺序访问一维数组元素的是 (D)

D 正确，在变址操作时，将计算机指令中的地址与变址寄存器中的地址相加，得到有效地址，指令提供数组首地址，由变址寄存器来定位数据中的各元素。所以它最适合按下标顺序访问一维数组元素，选D。

- A. 相对寻址
- B. 寄存器寻址
- C. 直接寻址
- D. 变址寻址

- 某指令功能为 $R[r2] \leftarrow R[r1] + M[R[r0]]$ ，其中两个源操作数分别采用寄存器，寄存器间接寻址方式，对于下列给定部件，该指令在取值及执行过程中需要用到的是 (B)

B 正确，指令译码器是在取指令的时候对指令进行译码时用到，在该指令取值以及执行过程中不发挥作用，它是在取指令时发挥作用。

I 通用寄存器组 II 算术逻辑单元 III 存储器 IV 指令译码器

- A. I、II、III、IV
- B. I、II、III
- C. II、III、IV
- D. I、III、IV

4. 零地址的运算类指令在指令格式中不给出操作数地址，参加运算的两个操作数来自 (C)

C 正确，零地址指令的运算属于堆栈的运算指令，参与操作的数据来自堆栈的栈顶和次栈顶。

拓展：零地址指令分为两种

1. 不需要操作数，如空操作，停机，关中断等指令
2. 堆栈计算机，两个操作数隐含存放在栈顶和次栈顶，计算结果压回栈顶。

- A. 累加器和寄存器
- B. 累加器和暂存器
- C. 堆栈的栈顶和次栈顶单元
- D. 暂存器和堆栈的栈顶单元

5. 下列关于一地址运算类指令的叙述中正确的是 (B)

B 正确，对于一地址指令来说可能只需要一个操作数，如加 1，减 1，取反，求补等。如 $OP(A1) \rightarrow A1$

也可能需要两个操作数，但其中一个操作数隐含在某个寄存器中（如隐含在 ACC 中）。

如 $(ACC) OP(A1) \rightarrow ACC$ 。ACC 中已经提前存好了另一个操作数。

- A. 仅有一个操作数，其地址由指令的地址码提供
- B. 可能有一个操作数，也可能有两个操作数
- C. 一定有两个操作数，另一个是隐含的
- D. 指令的地址码字段存放的一定是操作码

6. 在通用计算机指令系统的二地址指令中，操作数的物理位置可安排在 (D)

D 正确。

从操作数的物理位置来说，可把二地址指令可归结为三种类型：

存储器-存储器（**ss**）型指令：参与操作的数都放在内存里，从内存某单元中取操作数，操作结果存放至内存另一单元中。
因此机器执行这种指令需要多次访问内存。

寄存器-寄存器（**rr**）型指令：需要多个通用寄存器或个别专用寄存器，从寄存器中取操作数，把操作结果放到另一寄存器中。
机器执行这种指令的速度很快，不需要访问内存。

寄存器-存储器（**rs**）型指令：执行此类指令，既要访问内存单元，又要访问寄存器

- A. 两个数据寄存器
- B. 一个主存单元和一个数据寄存器
- C. 两个主存单元中
- D. 以上几项均可

7. 一个计算机系统采用 32 位单字长指令，地址码为 12 位，如果定义了 250 条二地址指令，那么单地址指令的条数有（D）

D 正确

指令单字长，则指令长度为32位

这个指令是从三地址扩展来的吗？不是，因为如果是三地址，则至少要 $12 \times 3 = 36$ 位
二地址要用掉 24 位地址码，操作码还剩下 $32 - 24 = 8$ 位，二地址指令可以有 256 条。

但是题目说只用了 250 条，也就是可以有 6 条用于扩展

而单地址指令，意味着地址码只占 12 位，相对于二地址指令，操作码又多出了 12 位。

所以单地址指令有： $6 \times 2^{12} = 24K$

- A. 4K
- B. 8K
- C. 16K
- D. 24K

8. 某计算机指令字长为 16 位，指令有双操作数、单操作数和无操作数 3 种格式，每个操作数字段均由 6 位二进制表示，该指令系统共有 m 条（ $m < 16$ ）双操作数指令，并存在无操作数指令。若采用拓展码技术，那么最多还可设计出单操作数指令的条数是（B）

B 正确，16位的指令，操作数为两个的时候(每个占6位)，那么操作码的长度为4位。

已知条件为双操作数指令为 m 条，还剩下 $2^4 - m$ 条双操作数指令没有用，可以用来向单操作数指令扩展。

因此最多单操作数指令为 $(2^4 - m) \times 2^6$ ，

如果不考虑有无操作数指令的话，那么就不用留下一个码点从单操作数指令扩展，上面就是最后的答案。但是题中要求存在无操作数指令，所以还要再减去一个码点，最后答案就了B

- A. 2^6
- B. $(2^4 - m) \times 2^6 - 1$

- C. $(2^4 - m) \times 2^6$
- D. $(2^4 - m) \times (2^6 - 1)$

9. 某计算机存储器按字（16）位编址，每取出一条指令后 PC 值加 1，说明其指令长度是（B）

B 正确，注意这边是按字编址，一个字为 16 位，PC + 1 说明变动了一个字长，一个字长为 2 字节，故选 B

- A. 1 个字节
- B. 2 个字节
- C. 3 个字节
- D. 4 个字节

10. 一条指令有 128 位，按字节编址，读取这条指令后，PC 的值应自动加（D）

D 正确，注意这里是按字节编址，其与上面一题不同，需要看变动多少个字节， $128 / 8 = 16$ 个字节，故选 D

- A. 1
- B. 2
- C. 4
- D. 16

11. 某计算机字长为 32 位，存储容量为 4MB，若按字编址，其寻址范围为（A）

A 正确，4MB 的 B 是字节 也就相当于 32Mb 这里的 b 是位
一个字 32 位，按字编址意思是每个字 (32 位) 分配一个地址，而要求的寻址范围就是一共能分成多少个地址。

所以答案是 $32\text{Mb} / 32\text{b} = 1\text{M} = 2^{20}$

- A. $0 \sim 2^{20} - 1$
- B. $0 \sim 2^{21} - 1$
- C. $0 \sim 2^{23} - 1$
- D. $0 \sim 2^{24} - 1$

12. 数据寻址和指令寻址的不同点在于（A）

A 正确，数据寻址寻找的是操作数的地址，指令寻址寻找的是下条指令的地址，它决定于程序转移地址。

拓展：寻址可以分为指令寻址和数据寻址。寻找下一条将要执行的指令地址称为指令寻址，寻找操作数的地址称为数据寻址。指令寻址比较简单，它又可以细分为顺序寻址和跳跃寻址。而数据寻址方式种类较多，其最终目的都是寻找所需要的操作数。

数据寻址和指令寻址只与寻找的地址类型有关，而与指令的长度无关。

- A. 前者决定操作数地址，后者决定程序转移地址
- B. 前者决定程序转移地址，后者决定操作数地址
- C. 前者是短指令，后者是长指令

- D. 前者是长指令，后者是短指令

13. 指令系统种采用不同寻址方式的主要目的是

- 缩短指令字长
- 扩大寻址范围
- 提高编程灵活性

14. 根据计算机指令的格式，可知指令执行过程中的操作数可能存放在 (D)

D 正确，在计算机指令的格式中，操作数通常分为立即数（存放在指令本身）、寄存器和存储器中的值。主存是主存储器，属于存储器。故选 D

拓展：在指令执行过程中，操作数通常会存放在寄存器中。这是由于寄存器具有更快的访问速度和更高的性能，相比于内存存储。在大多数处理器架构中，指令执行时会先将操作数从内存加载到寄存器中进行运算，然后再将结果写回内存。如寄存器寻址。

I 寄存器，II 指令本身，III 主存中，IV 控制存储器

- A. I, II
- B. II, III
- C. III, IV
- D. I, II, III

15. 在寄存器间接寻址方式中，操作数存放在 (D)

D 正确，注意这边说是寄存器间接寻址， $EA = R$ ，间接寻址就是说，放在寄存器的不是操作数，而是操作数的地址，操作数放在内存里面。

- A. 寄存器
- B. 堆栈
- C. 累加器
- D. 主存单元

16. 为了缩短指令中某个地址码的位数，同时使指令的执行时间又相对短，有效的寻址方式是 (B)

B 正确，寄存器寻址方式最显著的有点就是：

1. 由于寄存器数量较少，其地址码比主存单元地址短得多
2. 从寄存器中存取数据比从主存中快得多

- A. 立即寻址
- B. 寄存器寻址
- C. 直接寻址
- D. 寄存器间接寻址

17. 指令操作所需的数据不会来自 (D)

选择D, 见 14 题

- A. 寄存器

- B. 指令本身
- C. 主存
- D. 控制存储器

18. 下列有关变址寻址的说法不正确的是 (D)

选择D, 首先变址寻址中利用有效地址都加上了对应寄存器的内容, 扩大了指令的寻址范围。

D错误是因为变址寄存器的内容是由用户确定, 在执行过程中内容可变。

- A. 变址寻址扩大了指令的寻址范围
- B. 变址寻址适用于编制循环程序
- C. 变址寻址适合处理数组
- D. 变址寄存器的内容由操作系统确定, 在运行的过程中不可变

19. 试比较各种寻址方式的效率, 以下结论中不正确的是 (B)

选择B, 因为堆栈寻址可能是硬堆栈(寄存器), 或软堆栈(内存), 而采用软堆栈比寄存器寻址慢。

- A. 立即寻址的效率高于堆栈寻址
- B. 堆栈寻址的效率高于寄存器寻址
- C. 寄存器一次间接寻址的效率高于变址寻址
- D. 变址寻址的效率高于一次间接寻址

20. 设相对寻址的转移指令占两字节, 第一个字节是操作码, 第二个字节是相对位移量(用补码表示)。每当 CPU 从主存中取出第一个字节时, 即自动完成 $(PC) + 1 \rightarrow PC$ 。若当前 PC 的内容为 2000H, 要求转移到 200AH, 则该转移指令第二字节的内容应为 (B)

B 正确, 转移指令是 2 字节, 所以当 2000H 转移后变成 $2000 + 2 = 2002H$ 。因此 2002H, 转到200AH, 需要加 08H 即第二个字节相对位移量, 故选B。

- A. 07H
- B. 08H
- C. 09H
- D. 0AH

21. 程序控制类指令的功能是 (C)

C 正确, 程序控制类指令用于控制程序的执行顺序, 并使程序具有测试, 分析与判断的能力。

- A. 进行主存和 CPU 之间的数据传送
- B. 进行 CPU 和外设之间的数据传送
- C. 改变程序执行的顺序
- D. 控制进栈, 出栈操作

第三次 期末复习

1. 某计算机主存空间为 4GB 字长为 32 位，按字节编址采用 32 位定长指令格式。若指令按字边界对齐存放，则程序计数器(PC)和指令寄存器(IR)的位数至少分别是 (B)

B 正确，首先 $4GB = 2^{32}B$ ，字长为 32 位即 4B，说明一个指令占 4B，而 PC 的作用是定位到指令，而且这边是采用指令按字边界对齐，只需要每次变换找到下一条指令的位置即可， $4GB / 4B = 2^{30}$ ，所以可知 PC 只需要 30 位即可。

而指令寄存器(IR)存放的是正在执行的指令，由于每个指令都是 32 位，所以毫无疑问，IR 也是 32 位，故选 B。

- A. 30, 30
- B. 30, 32
- C. 32, 30
- D. 32, 32

2. 下列寄存器中，汇编语言程序员可见的是 (B)

B 正确，见第二章第八题，程序员可以通过制定待执行指令的地址来设置 PC 的值。

- A. 存储器地址寄存器(MAR)
- B. 程序计数器(PC)
- C. 存储器数据寄存器(MDR)
- D. 指令寄存器(IR)

3. 冯·诺依曼计算机中指令和数据均以二进制形式存放在存储器中，CPU 区分它们的依据是 (C)

C 正确，CPU 可以根据指令周期的不同阶段来区分是指令还是数据，通常在取指令阶段取出的是指令，其他阶段(分析取数阶段，执行阶段)取出的是数据。

- A. 指令操作码的译码结果
- B. 指令和数据的寻址方式
- C. 指令周期的不同阶段
- D. 指令和数据所在的存储单元

4. 下列关于程序计数器(PC)的叙述中，错误的是 (C)

C 错误，当执行到转移指令时，对于无条件转移或调用、返回等指令，则 PC 的值直接修改为目标指令地址；对于条件转移(分支)指令，则必须根据前面指令或当前指令执行的结果标志，确定是把转移目标地址还是把下条指令地址送到 PC。这边说总是，则不对。

- A. PC 中总是存指令地址
- B. PC 的值由 CPU 在执行指令过程中进行修改
- C. 转移指令时，PC 的值总是修改为转移目标指令的地址
- D. PC 的位数一般和存储器地址寄存器(MAR)的位数一样

5. 在一次无条件跳转指令的指令周期内，PC 的值被修改的次数是 (B)

B 正确，取指周期结束后，PC 的值 + 1；执行周期中，PC 值修改为要跳转的地址，故 PC 被修改两次。

- A. 1
- B. 2
- C. 3
- D. 无法确定

6. 数据寄存器中既能存放源操作数，又能存放结果的是 (C)

C 正确，累加器内容可以作为源操作数，也可以暂时存放 ALU 运算的结果信息。

- A. 锁存器
- B. 堆栈
- C. 累加器
- D. 触发器

7. 指令译码器是对 (B) 进行译码的

B 正确，指令的操作码字段表示该指令进行什么性质的操作，而指令译码器就是来翻译操作码，实现相关操作。

指令由操作码和地址码组成。

- A. 整条指令
- B. 指令的操作码字段
- C. 指令的地址
- D. 指令的操作数字段

8. 时钟周期：计算机中最小的时间单位，等于 CPU 主频的倒数。一个时钟周期内，CPU 仅完成一个最基本的动作

机器周期：机器周期又叫CPU周期。由若干时钟周期可以组成一个机器周期，机器周期就是要完成一个比较完整的子工作。计算机中为了方便管理，常把一条指令的执行过程划分为若干个阶段（如取指、间址、执行、中断等）每一阶段完成一个基本操作。注意：每一个基本操作都是由若干CPU最基本的动作组成。这个基本操作所需要的时间称为机器周期，则机器周期由若干个时钟周期组成。

指令周期：从取指开始到执行完成该指令所需要的全部时间。指令周期包含若干机器周期。

指令周期>机器周期>时钟周期

9. 在计算机体系结构中，CPU内部包括程序计数器PC、存储器数据寄存器MDR、指令寄存器IR和存储器地址寄存器MAR等。若CPU要执行的指令为:MOV R0, #100(即将数值100传送到寄存器R0中)，则CPU首先要完成的操作是 (C)

C正确，无论运行什么类型的指令，CPU首先需要取指令，取指令阶段的第一个操作就是将指令地址(程序计数器PC中的内容)送往存储器地址寄存器。取指周期完成的微操作序列是公共的操作，与具体指令无关，取指公共操作如下: (1) 将程序计数器PC中的内容送至存储器地址寄存器MAR，记作(PC)→MAR; (2) 向主存发读命令，记作Read; (3) 从主存中取出的指令送到存储器数据寄存器MDR，记作M(MAR)→MDR; (4) 将MDR的内容送至指令寄存器IR中，记作

(MDR)→IR; (5)将PC的内容递增,为取下一条指令做好准备,记作(PC)+1→PC。题中虽然给出了一条具体的指令“MOV R0, #100”,实际上CPU首先要完成的操作是取指令,与具体指令是没有关系的。

- A. 100→R0
- B. 100→MDR
- C. PC→MAR
- D. PC→IR

10. 取指令操作 (C)

这题不确定答案,知道答案可以和我联系。

- A. 受上一条指令操作码的控制
- B. 受当前指令操作码的控制
- C. 不受指令操作码的控制
- D. 受运算器中的条件码(或标志码)的控制

11. 有关指令周期叙述,错误的是 (B)

B 错误,很明显乘法比加法复杂,指令周期肯定更长一些。这题主要学习其他正确的选项。

- A. 指令周期的第一个阶段是取指令阶段
- B. 乘法指令和加法指令的指令周期一样长
- C. 一个指令周期由若干个机器周期或时钟周期组成
- D. 单周期处理器的指令周期就是一个时钟周期。

12. 在计算机中,存放微指令的控制存储器隶属于 (D)

C正确,CPU内部有一个控制存储器,里面存放着各种程序指令对应的微程序段。当CPU执行一句程序指令时,会从控制存储器里取一段与该程序指令对应的微程序解释执行,从而完成该程序语句的功能。CPU = 控制器 + 运算器 + (Cache)

- A. 外存
- B. 高速缓存
- C. 内存
- D. CPU

13. 在微程序控制中,操作控制信号将编成 (A) 形式

A 正确,微程序设计的概念是将一条机器指令编写成一个微程序,每一个微程序包含若干条微指令,每一条微指令对应一个或几个微操作命令。然后把这些微程序存放到一个控制存储器中,用寻找用户程序机器指令的方法来寻找每个微程序中的微命令

- A. 微指令
- B. 微地址
- C. 操作码

- D. 程序

14. 下列说法中正确的是 (D)

D 正确，一条微指令放在控制存储器的一个单元中。

A 错误，硬布线控制器采用硬件电路，速度快，但设计难度复杂，成本高。而微程序控制器速度慢，但灵活性高；

B 错误，控制存储器采用的是 ROM

C 错误，微指令计数器决定微指令执行顺序，指令执行顺序由 PC 决定。

- A. 采用微程序控制器是为了提高速度
- B. 控制存储器由高速 RAM 电路组成
- C. 微指令计数器决定指令执行顺序
- D. 一条微指令放在控制存储器的一个单元中。

15. 以下关于CPU的叙述中，错误的是 (B)

B 错误，CPU的功能主要包括程序控制、操作控制、时间控制和数据处理。CPU主要由运算器、控制器、寄存器组和内部总线等部件组成。CPU产生每条指令的操作信号并将操作信号送往相应的部件进行控制，因此说法A正确。

CPU中的控制器用于控制整个CPU的工作，它决定了计算机运行过程中的自动化，因此说法C正确。

程序计数器PC具有寄存信息和计数两种功能，又称为指令计数器。程序的执行分为两种情况，顺序执行和转移执行。在程序执行前，将程序的起始地址送入PC该地址在程序加载到内存时确定，执行指令时，CPU将自动修改PC的内容，当指令按照顺序执行时，PC加1。如果是转移指令，后继指令的地址根据当前指令的地址加上一个向前或向后转移的位移量得到。因此PC没有临时存储算术/逻辑运算结果的功能。因此说法B错误。

CPU中的控制器包括指令寄存器(IR)、程序计数器(PC)、地址寄存器(AR)和指令译码器(ID)。因此说法D正确。综上所述，答案为B。

- A. CPU产生每条指令的操作信号并将操作信号送往相应的部件进行控制
- B. 程序计数器PC除了存放指令地址，也可以临时存储算术/逻辑运算结果
- C. CPU中的控制器决定计算机运行过程的自动化
- D. 指令译码器是CPU控制器中的部件

16. 在微程序控制器中,机器指令与微指令的关系是 (B)

B 正确，一条机器指令的功能通常用许多条微指令组成的序列来实现，这个微指令序列称为微程序。

- A. 每一条机器指令由一条微指令来执行
- B. 一条机器指令由一段用微指令编成的微程序来解释执行
- C. 一段机器指令组成的程序可由一个微程序来执行
- D. 每一条微指令由一条机器指令来解释执行

17. 在微程序控制的计算机中，若要修改指令系统，只要 (D)

D 正确，在微程序控制的计算机中，若要修改指令系统，只需修改相应指令的微程序即可。这些微程序都存放在控制存储器中，所以只需改变控制存储器的内容。故选择 D

拓展：微程序控制器的设计思想和组合逻辑控制器的设计思想截然不同。它具有设计规整、调试维修以及更改、扩充指令方便的优点，易于实现自动化设计，已成为当前控制器的主流。但是，由于它增加了一级控制存储器，所以指令执行速度比组合逻辑控制器慢。

- A. 改变时序控制方式
- B. 改变微指令格式
- C. 增加微指令个数
- D. 改变控制存储器的内容

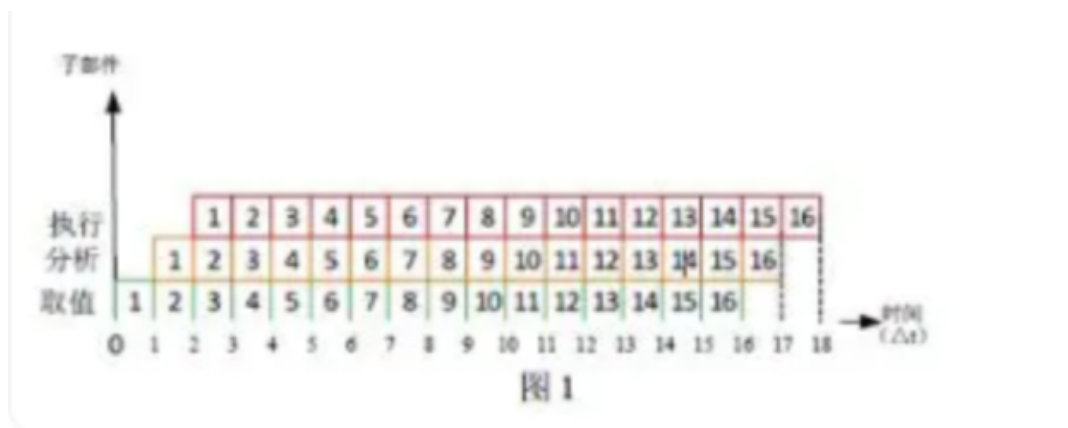
18. 下列关于并行微程序控制器的说法中,正确的是 (A)。

A 正确，并行微程序控制器中，在执行现行微指令的同时，取下一条微指令，故选择 A

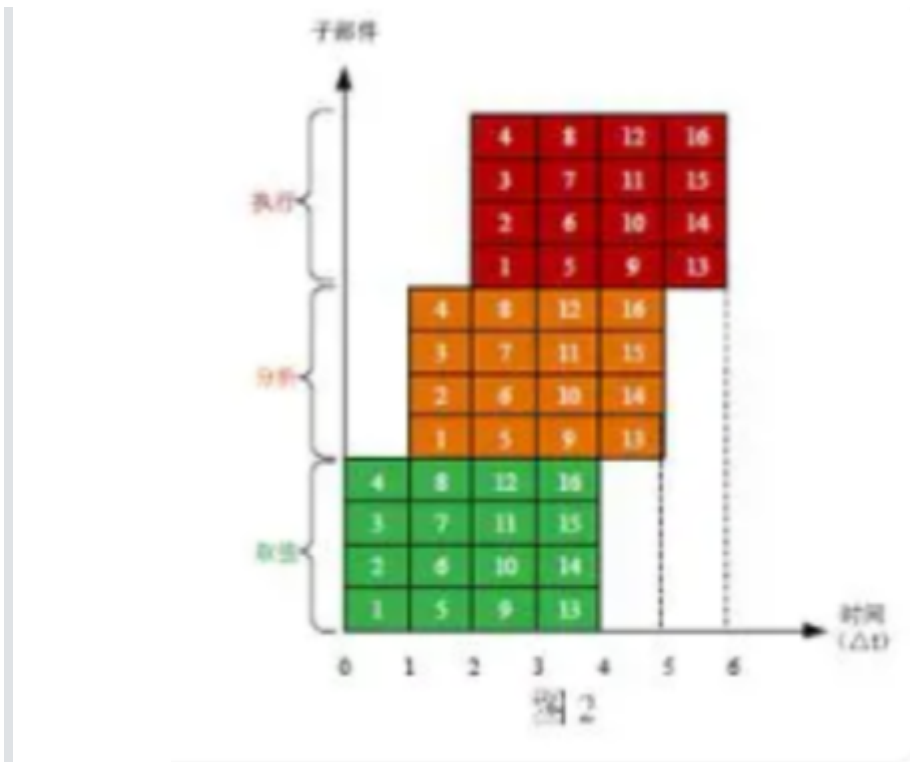
- A. 现行微指令的执行与取下一条微指令的操作并行
- B. 现行微指令的执行与取下一条微指令的操作串行
- C. 两条或更多微指令的执行在时间上并行
- D. 两条或更多微指令的取微指令操作在时间上并行

19. 设指令由取指，分析，执行 3 个子部件完成，并且每个子部件的时间均为 Δt ，若采用常规标量单流水线处理机 (即处理机的度为1)，连续执行 12 条指令，共需 (B)

B 正确，下图示意的是 16 条指令的流水线图，可以从中找到 12 处为 $14\Delta t$



拓展：若采用度为 4 的处理机，执行 16 条指令，需要时间为 $6\Delta t$ ，下图示意



- A. $12\Delta t$
- B. $14\Delta t$
- C. $16\Delta t$
- D. $18\Delta t$

20. 流水计算机中, 下列语句发生的数据相关类型是 (A)

ADD, R1, R2, R3, 即 $(R2) + (R3) \rightarrow R1$

ADD, R4, R1, R5, 即 $(R1) + (R5) \rightarrow R4$

A 正确, 在这两条指令中, 都对 R1 进行操作, 前面对 R1 写操作, 后面对 R1 读操作, 因此发生的是写后读。选择 A

- A. 写后读
- B. 读后写
- C. 写后写
- D. 读后读

21. 下列几项中, 流水线相关包括 (C)。

I. 结构相关 II. 数据相关 III. 指令相关 IV. 控制相关

C 正确, 流水线相关包括: 结构相关、数据相关和控制相关。

其中, 结构相关是由于不同指令欲同时使用一个部件而造成的

数据相关是指后面指令欲使用前面指令的结果时该结果还没有产生

控制相关是指指令执行的顺序发生改变而引起的流水线停顿以各类转移指令、分支指令以及中断或异常的出现都会引起控制相关

- A. I, II, III, IV
- B. I, II, III

- C. I, II, IV
- D. I, III, IV

22. 下列关于动态流水线和超标量处理器的说法中,错误的是(D)。

D 错误, 超标量技术指的是采用更多指令执行部件来构成多条流水线的技术, 而不是采用更多流水段。

- A. 超标量处理器中一定有多个不同的指令执行单元
- B. 动态流水线执行指令的顺序不一定是输入指令的顺序
- C. 超标量处理器不一定都采用动态流水线
- D. 超标量技术是指采用更多流水段个数的流水线技术

第四次 期末复习

1. 一次总线事务中, 主设备只需给出一个首地址, 从设备就能从首地址开始的若干连续单元读出或写入多个数据。这种总线事务方式称为 (C)

C 正确

猝发 (突发) 传输是在一个总线周期中, 可以传输多个存储地址连续的数据, 即一次传输一个地址和一批地址连续的数据,

并行传输是在传输中有多个数据位同时在设备之间进行的传输,

串行传输是指数据的二进制代码在一条物理信道上以位为单位按时间顺序逐位传输的方式,

同步传输是指传输过程由统一的时钟控制。

故选 C

- A. 并行传输
- B. 串行传输
- C. 突发传输
- D. 同步传输

2. 下列选项中, 可提高同步总线数据传输率的是 (B)

I. 增加总线宽度 II. 提高总线工作效率 III. 支持突发传输 IV. 采用地址/数据线复用

B正确,

I II 都很明显, 肯定增加的

III 突发传输一般表示的是两个设备之间进行数据传送的一种模式, 也可将其称为突发模式下的数据传输。而突发 (Burst) 是指在同一行中相邻的存储单元连续进行数据传输的方式, 连续传输的周期数就是突发长度 (Burst Lengths, 简称BL)。在突发传输模式下, 多个数据单元当做一个单元 (相当一个数据块) 来传送, 从而提高了传输效率。

IV 地址数据线复用并非提高而是降低, 毕竟同一时间你还得传地址过去, 等价的数据传输效率就低了。只是减少了成本。

- A. I、II

- B. I、II、III
- C. II、III
- D. I、II、III、和 IV

3. 下列关于总线设计的叙述中，错误的是 (A)

A 错误，

并行总线传输通常比串行总线传输速度快，但这不是绝对的。在实际时钟频率比较低的情况下，并行总线因为可以同时传输若干比特，速率确实比串行总线快。但是，随着技术的发展，时钟频率越来越高，并行导线之间的相互干扰越来越严重，当时钟频率提高到一定程度时，传输的数据已经无法恢复。而串行总线因为导线少，线间干扰容易控制，反而可以通过不断提高时钟频率来提高传输速率，A错误。

总线复用是指一种信号线在不同的时间传输不同的信息。可以使用较少的线路传输更多的信息，从而节省了空间和成本。故B正确。

突发（猝发）传输是在一个总线周期中，可以传输多个存储地址连续的数据，即一次传输一个地址和一批地址连续的数据，C正确。

分离事务通信即总线复用的一种，相比单一的传输线路可以提高总线的利用率，D正确。

- A.并行总线传输比串行总线传输速度快
- B.采用信号线复用技术可以减少信号线数量
- C.采用突发传输方式可提高总线数据传输速率
- D.采用分离事务通信方式可提高总线利用率

4. 下列选项中的英文缩写均为总线标准的是 (D)

D 正确，

A. PCI、**CRT**、USB、EISA

B. ISA、**CPI**、VESA、EISA

C. ISA、**SCSI**、**RAM**、**MIPS**

标黑的是不对的

目前典型的总线标准有：

ISA、EISA、VESA、PCI、PCI-Express、AGP、USB、RS-232C等。

微型机总线标准有下面几种：

ISA总线、EISA总线、MCA总线、VESA局部总线、PCI总线、AGP总线、USB总线、Alpha EV6总线、PCI局部总线、NGIO总线、IEEE1394、Future I/O总线

- A. PCI, CRT, USB, EISA
- B. ISA, CPI, VESA, EISA
- C. ISA, SCSI, RAM, MIPS
- D. ISA, EISA, PCI, PCI-Express

5. 下列关于 USB 总线特性的描述中，错误的是 (D)

D 错误, USB的特点有: (1) 即插即用 (2) 热插拔 (3) 很强的连接能力, 采用菊花链形式将总舵外设备连接起来 (4) 有很好的扩展性, 一个USB控制可扩充高达127个外部USB设备 (5) 告诉传输 速率可达480Mb/s (6) 串行总线, 一次只能传输一位数据

- A. 实现外设的即插即用和热插拔
- B. 可通过级联方式连接多台外设
- C. 是一种通信总线, 可连接不同外设
- D. 同时可传输 2 位数据, 数据传输率高

6. 系统总线中, 划分数据线, 地址线和控制线的根据是 (C)

C 正确, 系统总线按传送信息的不同划分位地址总线, 数据总线和控制总线。

- A. 总线所处的位置
- B. 总线的传输方向
- C. 总线的传输内容
- D. 总线的控制方式

7. 系统总线中地址线的作用是(C)

C正确, 地址线作用即为用于指定主存单元和I/O设备接口电路的地址

- A. 用于选择主存单元地址
- B. 用于选择进行信息传输的设备
- C. 用于指定主存单元和I/O设备接口电路的地址
- D. 用于传送主存物理地址和逻辑地址

8. 在系统总线中, 地址总线的位数 (B)

B 正确,

地址总线的位数与存储单元个数有关, 地址总线的位数越长, 可访问的存储单元个数就越多。

系统总线按传送信息的不同可以细分为:地址总线、数据总线和控制总线

地址总线由单方向的多根信号线组成, 用于CPU向主存、外设传输地址信息

数据总线由双方向的多根信号线组成, CPU可以沿这些线从主存或外设读入数据, 电可以沿这些线向主存或外设送出数据;

控制总线上传输的是控制信息, 包括CPU送出的控制命令和主存(或外设)返回CPU的反馈信号。

地址总线宽度决定了CPU可以访问的最大的物理地址空间, 简单地说就是CPU到底能够使用多大容量的主存。

例如, 32位地址线, 可寻址的最大容量为 $2^{32}=4096\text{MB}(4\text{GB})$ 。

- A. 与机器字长有关
- B. 与存储单元个数有关
- C. 与存储字长有关
- D. 与存储器带宽有关

9. 挂在总线上的多个部件 (B)

B 正确，总线是一组能为多个部件分时共享的信息传送线，用来连接多个部件并为之提供信息交换通路。所谓共享，指连接到总线上的所有部件都可通过它传递信息；分时性指某一时刻只允许一个部件将数据发送到总线上。因此，共享是通过分时实现的。

可以这么理解：可以一个人讲话，很多人听，但如果多个人同时说话就会很混乱，所以必须分时发送数据即，每次一个人说话传递信息，同时从总线接收数据，相当于听一个总览者讲话，下面人都可以同时听到。

- A、只能分时向总线发送数据，并只能分时从总线接收数据
- B、只能分时向总线发送数据，但可同时从总线接收数据
- C、可同时向总线发送数据，并同时从总线接收数据
- D、可同时向总线发送数据，但只能分时从总线接收

10. 间接寻址第一次访问内存所得到的操作数的有效地址，该地址经系统总线的 (A) 传送到 CPU。

A 正确，注意是第一次访问内存，由于是访问存储器得到的内容，所以是存储单元存放的数据，应该由数据总线传送给CPU。CPU经寻址方式判断后，再通过地址总线送给存储器，为读取操作数做准备。

- A. 数据总线
- B. 地址总线
- C. 控制总线
- D. 用户程序

11. 在单机系统中，三总线结构计算机的总线系统组成是 (C)

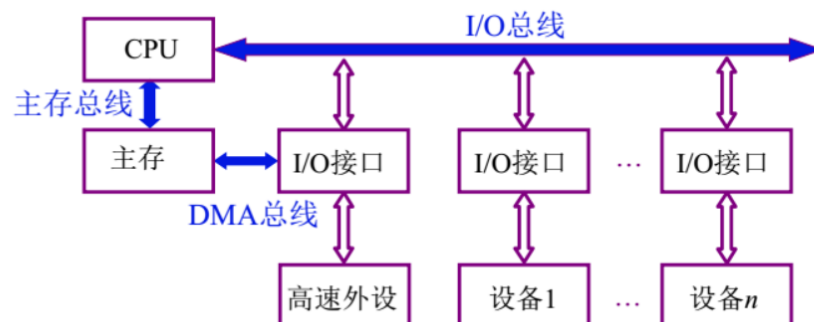
C 正确，在单机系统中,三总线结构的计算机的总线系统由系统总线、内存总线、I/O总线组成。

其中,系统总线用于CPU和内存之间传送地址、数据和控制信息;

内存总线使内存和高速外设之间能够直接传送数据

I/O总线提供CPU和各类外设之间的通信

拓展：三总线结构也可以由内存总线，I/O总线和DMA（直接内存访问（DMA，Direct Memory Access）总线组成



https://blog.csdn.net/qq_39368007

- A. 片内总线，系统总线和通信总线
- B. 数据总线，地址总线和控制总线

- C. 系统总线, 主存总线和 I/O 总线
- D. ISA总线, VESA 总线 和 PCI 总线

12. 微机中控制总线上完整传输的信号有 (B)

I 存储器和 I/O 设备的地址码; II 所有存储器和 I/O 设备的时序信号和控制信号; III 来自 I/O 设备和存储器的响应信号

B 正确, CPU 的控制总线提供的信号包括时序信号, I/O 设备和存储器的响应信号等。I 很明显是由地址总线传送的

- A. I
- B. II, III
- C. II
- D. I, II, III

13. 总线的从设备指的是 (B)

B 正确, 总线设备可分为主设备和从设备, 掌握总线控制权的设备是总线主设备, 而被主设备访问的设备是从设备

- A. 申请作为从设备的设备
- B. 被主设备访问的设备
- C. 掌握总线控制权的设备
- D. 总线源设备

14. 在集中式总线控制中, (C) 方式响应时间最快

C 正确, 独立请求方式的响应时间最快, 然而这是以增加控制线数和硬件电路为代价的。

- A. 链式查询
- B. 计数器定时查询
- C. 独立请求
- D. 不能确定哪一种

15. 在链式查询方式下, 若有 n 个设备, 则 (B)

B 正确, 链式查询方式采用菊花链的方式连接所有具有总线使用能力的器件, 各设备分别共有一根总线信号。

- A. 有 n 条总线请求
- B. 共有一条总线请求
- C. 有 $n - 1$ 条总线请求信号
- D. 无法确定

16. 为了对 n 个设备使用总线的请求进行仲裁, 在独立请求方式中需要使用的控制线数量为 (D)

D 正确,

对于n个设备而言,链式查询方式需要3条控制线;

计数器定时查询方式需要 $2 + \lfloor \log_2 n \rfloor$ 条控制线;

而独立请求方式需要 $2n+1$ 条控制线,包括 n 条总线请求控制线、 n 条总线批准控制线和1条总线忙控制线。

- A. n
- B. 3
- C. $2 + \lfloor \log_2 n \rfloor$
- D. $2n + 1$

17. 下列计算机总线属于串行总线的是 (B)

B 正确,

1394是高性能的串行总线。

IEEE-1394构建在菊花链或树状的拓扑结构上的,最新的高性能外部总线设计的趋势是使用**串行结构**,这样可以通过一根导线一次发送一位数据,而无须担心数据的到达时间,如IEEE-1394端口(使用高速串行技术)

选项A、C、D均属于并行总线。

- A. PCI
- B. 1394
- C. EISA
- D. ISA

18. 异步传输方式是指没有统一的时钟,也没有固定的时间间隔,完全依靠传送双方相互制约的“握手信号”来实

现定时控制。在下列各种情况中,最应采用异步传输方式的是(A)

A 正确,

I/O接口和打印机之间的速度差异较大,应采用异步传输方式来提高效率。异步定时方式能保证两个工作

速度相差很大的部件或设备之间可靠地进行信息交换

- A. I/O接口与打印机交换信息
- B. CPU与主存交换信息
- C. CPU和总线交换信息
- D. CPU内部的各个部件

19. 下列选择中,不属于 I/O 接口的是 (A)

A 正确,

A 不属于 I/O 接口。磁盘驱动器是用于读写磁盘上存储数据的设备。虽然它与I/O操作相关,但磁盘驱动器本身是存储设备。

B 属于 I/O 接口。打印机适配器是一种用于连接计算机和打印机之间的接口。它负责转换计算机系统中的数据格式和打印机所需的格式,并管理数据的传输和控制打印机操作。

C 属于 I/O 接口。网络控制器是用于连接计算机系统与计算机网络之间的接口。它负责协调和管理数据在计算机和网络之间的传输，执行网络协议和管理网络连接。

D 属于 I/O 接口。可编程中断控制器是计算机系统的一种设备，用于管理和处理系统中的中断信号。它负责识别中断来源、优先级和中断处理程序的分发。

- A. 磁盘驱动器
- B. 打印机适配器
- C. 网络控制器
- D. 可编程中断控制器

20. 下列选择中，在 I/O 总线的数据线上传输的信息包括 (D)

I. I/O 接口中的命令字 II. I/O 接口中的状态字 III. 中断类型号

D 正确，在 I/O 总线的数据线上传输的信息包括 I/O 接口中的命令字，状态字以及真正的数据，而中断类型号也是通过数据线传输的。

- A. I, II
- B. I, III
- C. II, III
- D. I, II, III

21. 下列有关 I/O 接口的叙述中错误的是 (D)

D 错误，采用统一编码方式，存储器和 I/O 端口共用统一的地址空间，不需要专用的 I/O 指令，任何对存储器数据进行操作的指令都可用于 I/O 端口的数据操作。

- A. 状态端口和控制端口可以合用同一寄存器中
- B. I/O 接口中 CPU 可访问寄存器，称为 I/O 端口
- C. 采用独立编址方式时，I/O 端口地址和主存地址可能相同
- D. 采用统一编址方式时，CPU 不能用访存指令访问 I/O 端口

22. I/O 指令实现的数据传送通常发生在 (D)

D 正确，

A. I/O 设备和 I/O 端口之间。这不是通过 I/O 指令直接实现的，而是通过系统的其他部分(例如 DMA 或中断)。

B. 通用寄存器和 I/O 设备之间。数据不会直接从通用寄存器传输到 I/O 设备，它们需要一个中介，通常是 I/O 端口。

C. I/O 端口和 I/O 端口之间。这不是常规的操作，两个 I/O 端口之间通常不会直接传输数据。

D. 通用寄存器和 I/O 端口之间。这是正确的选择。当 CPU 需要与外部设备进行通信时，它通常将数据从通用寄存器写入 I/O 端口，或从 I/O 端口读取数据到通用寄存器。

答案是 D。数据传送通常发生在通用寄存器和 I/O 端口之间。

- A. I/O 设备和 I/O 端口之间
- B. 通用寄存器和 I/O 设备之间
- C. I/O 端口和 I/O 端口之间
- D. 通用寄存器和 I/O 端口之间

23. 下列事件中, 属于外部中断事件的是 (C)

I. 访存时缺页 II. 定时器到时 III. 网络数据包到达

C 正确, 访存时缺页属于内部中断。

- A. I, II
- B. I, III
- C. II, III
- D. I, II, III

24. 在采用中断I/O方式控制打印输出的情况下, CPU和打印控制接口中的 I/O 端口之间交换的信息不可能是 (B)

选择 B,

I/O 接口的功能包括:①选址功能;②传送命令功能;③传送数据功能;④反映I/O 设备工作状态功能。A项为数据, C项为设备状态, D项为命令。B项, 主存地址在中断方式控制下是不需要的, 因此, 它不可能是 CPU 和打印控制接口中的I/O 端口之间交换的信息。

- A.打印字符
- B.主存地址
- C.设备状态
- D.控制命令

25. 下列选四项中,能引起外部中断的事件是(A)。

A 正确, 浮点数下溢一般做“机器零”处理,不引起中断;除数为0、访存缺页会引出内部中断;只有键盘输入能引起外部中断

- A.键盘输入
- B. 除数为0
- C.浮点运算下溢
- D. 访存缺页

26. 响应外部中断的过程中,中断隐指令完成的操作,除保护断点外,还包括 (B)

I. 关中断 II. 保存通用寄存器的内容 III. 形成中断服务程序入口地址并送PC

B 正确, 中断隐指令完成以下三大任务: 程序断点进栈、关中断、向量地址-PC

- A. 仅I、II
- B. 仅I、III
- C. 仅II、III
- D. I、II、III

27. 单级中断系统中,中断服务程序内的执行顺序是 (A)

I.保护现场 II. 开中断 III.关中断 IV.保存断点 V.中断事件处理 VI.恢复现场 VII.中断返回

A 正确, 在单级中断系统中, 一旦响应了一个中断, 可以不必使用关中断的措施来防止再来一个新的
中断干扰已经开始的中断保存现场和恢复现场的工作。响应与处理中断的几项操作是有严格顺序关系的,
不能颠倒。

- A. I→V→VI→II→VII
- B. III→I→V→VII
- C. III→IV→V→VI→VII
- D. IV→I→V→VI→VII

28. 下列关于外部I/O中断的叙述中,正确的是(C)。

C 正确,
A 项, 中断控制器按中断请求的重要性和紧迫程度进行中断优先级排队, 与先后次序无关。
B项, CPU 响应中断时, 通过执行保存寄存器的数值(保护现场)完成通用寄存器的保护。
D项, 有中断请求时, 需要进行进程优先级的比较, CPU 不是立即暂停当前指令执行, 转去执行中断服务程序。

- A. 中断控制器按所接受中断请求的先后次序进行中断优先级排队
- B. CPU响应中断时,通过执行中断隐指令完成通用寄存器的保护
- C. CPU只有在处于中断允许状态下时,才能响应外部设备的中断请求
- D. 有中断请求时,CPU立即暂停当前指令执行,转去执行中断服务程序

29. 系统将数据从磁盘读到内存的过程包括以下操作

- ① DMA控制器发出中断请求
- ② 初始化DMA控制器并启动磁盘
- ③ 从磁盘传输一块数据到内存缓冲区
- ④ 执行“DMA结束”中断服务程序

正确的执行顺序是()。

B 正确, 系统将数据从磁盘读到内存的过程:在开始DMA传输时, 主机向内存写入DMA命令块
向DMA控制器写入该命令块的地址, 启动I/O设备;CPU继续其他工作, DMA控制器则继续下去
直接操作内存总线, 将地址放到总线上开始传输;完成整个传输, DMA控制器中断CPU。

- A. ③ →① →② →④
- B. ② →③ →① →④
- C. ②→① →③ →④
- D. ① →② →④ →③

30. 一台显示器的图形分辨率为1024x768, 要求显示256种颜色, 显示存储器VRAM的容量至少为 (B)

B 正确, $N = \text{VRAM容量} / (\text{垂直点数} \times \text{水平点数})$ 其中, N为显示每点颜色的二进制[2位数, 本题
中256种颜色故需8位, $N=8$, $\text{VRAM容量} = (1024 \times 768) \times 8 = 768\text{KB}$, 故至少需要 1MB

- A. 512KB

- B. 1MB
- C. 3MB
- D. 4MB

31. 为提高存储器存取效率,在安排磁盘上的信息分布时,通常是 (C)

C 正确, 如果选择同一柱面上的不同磁道,由于各记录面的磁头已同时定位,换道的时间只是磁头选择电路的译码时间,相对于定位操作可以忽略不计.

- A.存满一面,再存另一面
- B.尽量将同一文件存放在一个扇区或相邻扇区的各磁道上
- C.尽量将同一文件存放在不同面的同一磁道上
- D.上述方法均有效

32. I/O 采用统一编址时, 进行输入输出操作的指令是 (B)

B 正确, 统一编址就是将 I/O 地址看作是存储器地址的一部分, 无须专用的 I/O 指令, 因此进行输入输出操作时使用访存指令。

- A. 控制指令
- B. 访存指令
- C. 输入输出指令
- D. 以上都不对

33. 将外设与主存统一编址, 一般是指 (C)

C 正确, 统一编址时把 I/O 接口中的端口(有关寄存器)作为主存单元进行访问,通常每个端口占一个主存单元地址

- A.每台设备占一个地址码
- B.每个外设接口占一个地址码
- C.接口中的有关寄存器各占一个地址码
- D.每台外设由一个主存单元管理

34. 在计算机系统中, 外部设备通过 I/O 接口与主板的系统总线相连接, 其功能是(D)

D 正确, 这些都是 I/O 接口的功能

- A.数据缓冲和数据格式转换
- B.检测外部设备的状态
- C.控制外部设备的操作
- D.前三种功能的综合作用

35. I/O接口中数据缓冲器的作用是 (A)

A 正确, I/O接口中数据缓冲器的作用是用来暂存外设和CPU之间传送的数据, 当输入时, 由设备送往数据缓冲器, 再由数据缓冲器通过数据总线送到内存。反之, 输出时, 由内存通过数据总线送到数据缓冲器, 然后再送到设备。

- A. 用来暂存外设和CPU之间传送的数据
- B. 用来暂存外设的状态
- C. 用来暂存CPU发出的命令
- D. 以上都是

36. 下面关于 I/O 设备控制方式的叙述中, 正确的是 (C)

C 正确,

A. 程序查询方式的 CPU 效率较高 - 错误。程序查询方式需要 CPU 持续轮询 I/O 设备状态, 占用 CPU 时间较多, 效率较低。

B. 对于批量数据传输, 通常需要进行数据流控制, 以确保发送方和接收方之间的数据传输速率匹配。使用中断可能会使数据流控制更加复杂, 因为中断的发生和处理可能会影响数据传输的速率和连续性。

D. DMA 方式完成 I/O 处理较慢 - 错误。DMA (Direct Memory Access) 方式可以在不占用 CPU 的情况下直接在内存和 I/O 设备之间传输数据, 效率较高。

- A. 程序查询方式的 CPU 效率较高
- B. 中断方式适于批量数据传送
- C. 中断方式需要 CPU 的较多干预
- D. DMA 方式完成 I/O 处理较慢

37. 当有中断源发出请求时, CPU可执行相应的中断服务程序。提出中断请求的可以是 (C)

C 正确, 中断请求可以来自 CPU 外部, 也可以来自 CPU 内部。A,B,D 都不能发出中断请求

- A. 通用寄存器
- B. 专用寄存器
- C. 外部事件
- D. Cache

38. 在中断周期, CPU主要完成以下工作 (A)

A正确, 在中断周期CPU执行中断隐指令, 完成关中断、保存断点、形成中断服务程序入口地址3项操作。

- A. 关中断, 保护断点, 发中断响应信号并形成中断服务程序入口地址
- B. 开中断, 保护断点, 发中断响应信号并形成中断服务程序入口地址
- C. 关中断, 执行中断服务程序
- D. 开中断, 执行中断服务程序

39. 在CPU响应中断时, 保护两个关键的硬件状态是(B)。

B 正确，程序计数器(PC)的内容是被中断程序尚未执行的指令地址，程序状态字(PSW)寄存器保存各种状态信息。CPU响应中断后，需要保护中断的CPU现场，将PC和PSW压入堆栈，这样等到中断结束后，就可以将压入堆栈的原PC和PSW的内容返回相应的寄存器，原程序从断点开始继续执行。

- A.PC和IR
- B.PC和PSW
- C.AR和IR
- D.AR和PSW

40. 下面关于中断的顺序,排列正确的是(A)

A 正确，中断的顺序是中断请求、中断响应和中断处理。

- A.中断请求、中断响应、中断处理
- B.中断响应、中断请求、中断处理
- C.中断处理、中断响应、中断请求
- D.中断响应、中断处理、中断请求

41. I/O与主机交换信息的方式中，中断方式的特点是(B)。

B 正确，当有中断发生时，CPU将暂停当前的主程序，转去处理中断事务，中断处理"完后，继续原来暂停的主程序。因此，中断方式的特点是:(1)CPU的效率 high，不必查询等待;(2)**CPU与外设可并行工作**;(3)外设具有申请服务的主动权。

在I/O与主机交换信息时，传送与主程序串行工作。

- A.CPU与设备串行工作，传送与主程序串行工作
- B.CPU与设备并行工作，传送与主程序串行工作
- C.CPU与设备并行工作，传送与主程序并行工作
- D.CPU与设备串行工作，传送与主程序并行工作

42. 中断允许触发器用于(C)

C 正确，中断允许触发器的作用是控制是否允许中断。当中断允许触发器为0时,中断关闭(关中断),所有中断源的中断请求都不能得到响应;当中断允许触发器为1时,中断允许(开中断),来自中断源的中断请求可以得到响应。

- A.向CPU发中断请求
- B.指示正有中断在进行
- C.开放或关闭中断系统
- D.指示中断处理结束

43. 若有多个中断同时发生,系统将根据中断优先级响应优先级最高的中断请求。若要调整中断事件的处理次序,可以利用(D)

D 正确，如果有多个中断同时发生,要调整中断事件的处理次序,可以利用中断屏蔽的方式

- A. 中断嵌套

- B. 中断向量
- C. 中断响应
- D. 中断屏蔽

44. 设置中断屏蔽标志可以改变(D)。

D 正确，中断屏蔽标志的二种作用是实现中断升级，即改变中断处理的次序(注意分清中断响应次序和中断处理次序)

中断响应次序由硬件排队电路决定。当中断请求先后出现时，允许优先级高的中断请求打断优先级低的中断服务程序。

- A.多个中断源的中断请求优先级
- B.CPU对多个中断请求响应的优先次序
- C.多个中断服务程序开始执行的顺序
- D.多个中断服务程序执行完的次序