

## 例题1



### 8.2 试题精解

#### 例题1 ( 2005年11月试题1 )

阵列处理机属于      计算机。

- A.SISD B.SIMDC.MISDD.MIMD

#### 试题分析

Flynn分类法将计算机系统结构分为以下4种：

SISD:单指令流单数据流；

SIMD:单指令流多数据流；

MISD:多指令流单数据流；

MIMD:多指令流多数据流。

根据阵列机的定义，它将大量重复设置的处理单元互连构成阵列，在单一控制部件的控制下，向各处理单元分配各自的数据，以达到并行执行同一条指令的目的。因此，阵列处理机是单指令流多数据流 ( SIMD ) 计算机。

#### 试题答案

**B**

版权方授权希赛网发布，侵权必究

[上一节](#)      [本书简介](#)      [下一节](#)

## 例题2

#### 例题2 ( 2005年11月试题2 )

采用\_\_\_\_\_不能将多个处理机互连构成多处理机系统。

- A.STD总线 B.交叉开关 C.PCI总线 D.Centronic总线

#### 试题分析

STD总线即STD bus ( Standard for an 8-bit microcomputer bus system )，是一种规模最小、面向工业控制的8位系统总线，支持多处理器系统。1978年Pro-Log公司将STD总线作为工业标准推出，随后被批准为国际标准IEEE 961，是一种很老的总线。按STD总线标准设计的模块式工控机，采用小板结构，每种模板功能单一，有CPU模板、内储模板、键盘显示模板、串行接口模板、A/D转换模板、D/A转换模板等，按扩展要求可选用其中几块模板，并支持多个CPU模板，非常灵活、方便、可靠。

交叉开关将各个CPU连接成动态互连网络，组成多处理机系统。

PCI ( Peripheral Component Interconnect, 外部组件互连 ) 总线用于将显卡、声卡、网卡和硬盘控制器等高速外围设备直接挂在CPU总线上，其负责CPU和外围设备的通信。

集群系统是一种多处理器系统。集群系统一般使用局域网将一组高性能工作站或者高档PC按一定结构连接起来，并在并行程序设计及可视化人机交互集成开发环境支持下，统一调度、协同处理，实现高效并行处理。

我们知道，通过插在PCI插槽上的网卡可以组建局域网，所以，通过PCI总线是可以组建多处理器系统。

Centronic总线属于外部总线，它的接口是一种打印机并行接口标准，用于将计算机与打印机等外设相连接。

### 试题答案

C

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第8章：计算机组成与体系结构

作者：希赛教育软考学院 来源：希赛网 2014年02月10日

## 例题3

### 例题3 ( 2005年11月试题4, 5 )

每一条指令都可以分解为取指、分析和执行三步。已知取指时间 $t_{取指}=5Dt$ , 分析时间 $t_{分析}=2Dt$ , 执行时间 $t_{执行}=5Dt$ . 如果按顺序方式从头到尾执行完500条指令需

( 4 )  $Dt$ . 如果按照[执行] $k$ 、[分析] $k+1$ 、[取指] $k+2$ 重叠的流水线方式执行指令，从头到尾执行完500条指令需 ( 5 )  $Dt$ .

( 4 ) A.5590 B.5595 C.6000 D.6007

( 5 ) A.2492 B.2500 C.2510 D.2515

### 试题分析

按顺序方式执行指令，每条指令从取指到执行共耗时 $12Dt$ , 所以500条指令共耗时： $12 \times 500 = 6000Dt$ .

采用流水线方式时，系统在同一时刻可以进行第 $k$ 条指令的取指，第 $k+1$ 条指令的分析，第 $k+2$ 条指令的执行，所以效率大大提高了。采用流水线的执行示意图如图8-1所示。

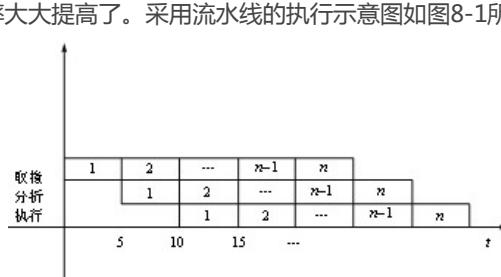


图8-1 流水线执行示意图一

平时大家看到的都是这样的示意图，但是平时我们看到的图都是笼统的。这里把所有周期都定为统一长度，这样流水线的总时间为： $(n+2)$ 周期。如此题中为： $(500+2) \times 5 = 2510$ 。

但我们平时用的流水线计算公式是：第一条指令顺序执行时间+ (指令条数-1) x周期这个公式

是怎么来的呢？

请大家看图8-2。

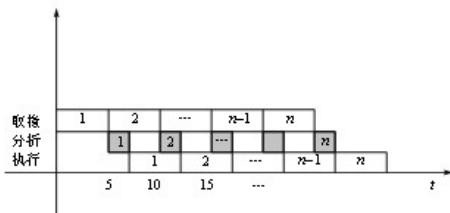


图8-2 流水线执行示意图二

对于此题而言，关键在于指令的分析时间，周期是5D，而实际完成分析只需要2D时间，所以正常运行时空图应如图8-2所示，其中黑色块是分析的真实发生时间。所以采用流水线的耗时为：

5+2+5? ( 500-1 ) +5=2507。

题目是按图8-1来计算的，计算结果为2510。

### 试题答案

C C

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第8章：计算机组成与体系结构

作者：希赛教育软考学院 来源：希赛网 2014年02月10日

### 例题4

#### 例题4 ( 2006年5月试题1 )

两个同符号的数相加或异符号的数相减，所得结果的符号位SF和进位标志CF进行\_\_\_\_\_运算为1时，表示运算的结果产生溢出。

- A.与 B.或 C.与非 D.异或

#### 试题分析

解答这道题，我们首先需要清楚一个概念，即什么是溢出。溢出，是指运算结果超出机器数的表示范围。从这个概念我们可以了解到，两个异号数相加不会产生溢出，仅两个同号数相加时才有可能产生溢出。两个正数相加而绝对值超出允许的表示范围时称为正溢，两个负数相加而绝对值超出允许的表示范围时则称为负溢。一旦溢出，溢出的部分将丢失，留下的结果将不正确。如果只有一个符号位，溢出将使结果的符号位产生错乱。因此，一般计算机中都设置了溢出判断逻辑，如果产生溢出，将停机并显示“溢出”标志。

现在我们来看几个典型的例子，从中我们可以总结出判断溢出的方法。其实在平时的学习当中，大家也应该尽可能多地自己总结规律，而不要光看书上的判断方法。

在下面的例题中，二进制数首位为符号位，后面4位为数据位。采用补码运算。

例1：

$$3 + 6 = 9$$

$$0\ 0011$$

$$0\ 0110$$

0 1001

例2:

$8+9=17$

0 1000

0 1001

1 0001 ( 正溢 )

例3:

$8 + (-5) = 3$

0 1000

1 1011

0 0011

例4:

$(-9) + (-8) = -17$

1 0111

1 1000

0 1111 ( 负溢 )

看完上面的4个运算式，我们可以开始总结规律了。由于上面的计算，是对两个4位的带符号二进制数进行运算，运算结果仍是一个4位带符号二进制数。所以其运算结果的范围应是： $-16 \sim +15$ ，非常明显，上面的例2和例4的结果溢出了。接下来，我们对这几个例题进行详细分析。为了便于分析，我们令两个操作数的符号位分别为： $S_a$ 和 $S_b$ 。结果的符号位为： $S_f$ 。符号位直接参与运算，所产生的符号位进位为 $C_f$ 。将符号位之后的 $A_1$ 和 $B_1$ 称为最高有效位，它产生的进位为 $C$ 。在例3中， $C=1$ ，但并未溢出，所以进位不等于溢出，不能简单地根据单个进位信号去判断有无溢出，而应当从几个相关信号之间的关联去进行溢出判断。

根据这些信号的关联，可以推出多种判断溢出的关系。

溢出判断方法一：

溢出

这个式子其实是由两部分组成的，分别说明了两种情况的溢出。

第一种情况：表示当操作数符号位 $S_a$ 和 $S_b$ 都为0，且结果的符号位为1时，产生溢出。符号位 $S_a$ 和 $S_b$ 都为0表示两个操作数均为正数，所以这种情况被称为“正溢”。例2符合此情况。

第二种情况：表示当操作数符号位 $S_a$ 和 $S_b$ 都为1，且结果的符号位为0时，产生溢出。符号位 $S_a$ 和 $S_b$ 都为1表示两个操作数均为负数，所以这种情况被称为“负溢”。例4符合此情况。

溢出判断方法二：

溢出

这种方法是从两种进位信号之间的关联角度出发的， $C_f$ 为符号运算后产生的进位， $C$ 为最高有效数位产生的进位。分析前面的几个例题会发现：产生正溢时，由于操作数较大，因而 $C=1$ ，但由于两个正数的符号位都为0，则 $C_f=0$ ；产生负溢时，由于补码映射值较小，因而 $C=0$ ，但由于两个负数的符号位皆为1，则 $C_f=1$ ；其他未溢出情况， $C_f$ 与 $C$ 都相同，所以得到第二种判断逻辑，即当 $C_f$ 与 $C$ 不同时表明溢出。这是在单符号位补码中应用较多的判断逻辑。

溢出判断方法三：

单符号位的信息量只能表示两种可能：数为正或为负，如果产生溢出，就会使符号位的含义产生混乱。将符号位扩充为两位，信息量扩大，就能判断是否有溢出及结果的正确符号。同样以前面的几个算式为例。

例5：

$3+6=9$

00 0011

00 0110

00 1001

例6：

$8+9=17$

00 1000

00 1001

01 0001 ( 正溢 )

例7：

$8+(-5)=3$

00 1000

11 1011

00 0011

例8：

$(-9)+(-8)=-17$

01 0111

01 1000

10 1111 ( 负溢 )

通过上面的例子，可定义符号位的含义为：

00--结果为正，无溢出；

01--结果正溢；

10--结果负溢；

11--结果为负，无溢出。

分析到这里，我们发现此题最适合用最后一种方法来解答。在双符号位中，高位就是符号位的进位CF；而低位就是结果的符号位SF.我们可以看到，当CF与SF不同时，表示溢出；而相同时，表示操作正常。能达到此效果的运算为"异或"运算，所以此题选择D.

### 试题答案

D

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

### 例题5 ( 2006年5月试题3, 4 )

高速缓存Cache与主存间采用全相联地址映像方式，高速缓存的容量为4MB,分为4块，每块1MB,主存容量为256MB.若主存读写时间为30ns,高速缓存的读写时间为3ns,平均读写时间为3.27ns,则该高速缓存的命中率为 ( 3 ) %.若地址变换表如表8-2所示，则主存地址为8888888H时，高速缓存地址为 ( 4 ) H.

表8-2 地址变换表

0	38H
1	88H
2	59H
3	67H

- ( 3 ) A.90      B.95      C.97      D.99  
( 4 ) A.488888      B.388888      C.288888      D.188888

#### 试题分析

第 ( 3 ) 空是一个简单的计算题。我们设高速缓存的命中率为 : t.

则

$$30? ( 1-t ) + 3xt = 3.27$$

解方程得 :  $t=0.99$ .所以高速缓存的命中率为99%.

接下来看第 ( 4 ) 空，由于高速缓存的容量为4MB,分为4块。所以把高速缓存的22位长地址划分为两部分，块号为2位，而块内地址为20位。主存容量为256MB,所以主存地址长度为28位。这样主存的块号为8位，块内地址为20位。此时我们先将主存地址8888 888H化为二进制数 : 1000 1000 1000 1000 1000 1000,其中斜体为块号 : 88H,加粗部分为块内地址 : 88888.查表得到Cache对应块号为1H,所以高速缓存地址为188888H.所以答案为D.

#### 试题答案

- ( 3 ) D ( 4 ) D

版权方授权希赛网发布，侵权必究

上一节      本书简介      下一节

第 8 章 : 计算机组成与体系结构

作者 : 希赛教育软考学院   来源 : 希赛网   2014年02月10日

### 例题6

#### 例题6 ( 2006年5月试题6 )

某指令流水线由5段组成，各段所需要的时间如图8-3所示。

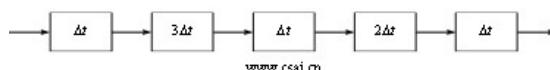


图8-3 流水线各段时间示意图

连续输入10条指令时的吞吐率为\_\_\_\_\_。

- A.10/70Dt   B.10/49Dt   C.10/35Dt   D.10/30Dt

#### 试题分析

要解此题，必须先清楚吞吐率的概念。吞吐率是指在流水线单位时间内能流出的任务数。现在

任务数是10,我们只要求出完成10个任务所需要的时间,就可以得出吞吐率了。

$t = \text{执行第1条指令所需时间} + (\text{指令条数}-1) \times \text{流水线周期}$

$$= (1^{\Delta t} + 3^{\Delta t} + 1^{\Delta t} + 2^{\Delta t} + 1^{\Delta t}) + (10-1) \times 3^{\Delta t}$$

$$= 35^{\Delta t}$$

所以

$$\text{吞吐率} = 10/35^{\Delta t}.$$

### 试题答案

C

版权方授权希赛网发布,侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第8章:计算机组成与体系结构

作者:希赛教育软考学院 来源:希赛网 2014年02月10日

## 例题7

### 例题7(2006年11月试题1)

若内存按字节编址,用存储容量为32K?8比特的存储器芯片构成地址编号为A0000H~DFFFFH的内存空间,则至少需要\_\_\_\_\_片。

- A.4 B.6 C.8 D.10

#### 试题分析

此题的解题思路是先计算出地址编号为A0000H~DFFFFH的内存空间大小,然后用空间大小除以芯片容量,得到芯片数量。在这个操作过程中,运算单位及数制的一致性特别需要注意,在进行运算之前,一定得把单位转化成相同的。下面是具体运算过程。

$DFFFFH - A0000H + 1 = 40000H$ ,转化为十进制为218.由于内存是按字节编址,所以空间大小应为28KB,即256KB,32K?8比特的芯片即32K?1字节的芯片,所以 $256KB / 32KB = 8$ .所以正确答案为C.

### 试题答案

C

版权方授权希赛网发布,侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第8章:计算机组成与体系结构

作者:希赛教育软考学院 来源:希赛网 2014年02月10日

## 例题8

### 例题8(2006年11月试题3)

设指令由取指、分析、执行3个子部件完成,每个子部件的工作周期均为Dt.采用常规标量单流水线处理机,若连续执行10条指令,则共需时间\_\_\_\_\_Dt。

- A.8 B.10 C.12 D.14

## 试题分析

设流水线由m段组成，每段所需时间分别为 $\Delta t_i$  ( 1 ( i ( m ) )，完成n个任务的实际时间可计算如下： $\sum_{i=1}^m \Delta t_i + (n-1)\Delta t_j$  (  $\Delta t_i$  为最慢一段所需时间 )

本题中流水线每一段的工作周期均为Dt,所以流水线的操作周期也就是Dt,所以总时间为 ( Dt + Dt + Dt ) + ( 10-1 ) x Dt =3+9=12.

## 试题答案

C

版权方授权希赛网发布，侵权必究

上一节 本书简介 下一节

第 8 章：计算机组成与体系结构

作者：希赛教育软考学院 来源：希赛网 2014年02月10日

## 例题9

### 例题9 ( 2006年11月试题4, 5 )

某计算机的时钟频率为400MHz,测试该计算机的程序使用4种类型的指令。每种指令的数量及所需指令时钟数 ( CPI ) 如表8-3所示，则该计算机的指令平均时钟数约为 ( 4 ) ;该计算机的运算速度约为 ( 5 ) MIPS.

表8-3 指令情况表

指令类型	指令数目(条)	每条指令所需时钟数
1	160000	1
2	30000	2
3	24000	4
4	16000	8

( 4 ) A.1.85 B.1.93 C.2.36 D.3.75

( 5 ) A.106.7 B.169.5 C.207.3 D.216.2

## 试题分析

此题第 ( 4 ) 空是求计算机的指令平均时钟数。现已知计算机共有4类指令，并且知道每类指令的数量和所需时钟数，所以只需要计算出所有指令所需时钟数及指令总数，两者相除则可得到平均时钟数。

指令总数为：

$$160000+30000+24000+16000 = 230000$$

所有指令所需时钟数为：

$$160000 \times 1 + 30000 \times 2 + 24000 \times 4 + 16000 \times 8 = 444000$$

所以平均时钟数为：

$$444000 / 230000 = 1.93$$

接下来求计算机的运算速度是1秒钟执行多少条指令。现已知计算机的时钟频率为400MHz,每执行一条指令平均需要1.93个时钟，所以用时钟频率400MHz除以1.93即可得到运算速度。

$$400MHz / 1.93 = 207.2539 MIPS, 所以此题答案为：B,C.$$

### 例题10

#### 例题10（2006年11月试题6）

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，每个操作数字段均用6位二进制数表示，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。若采用扩展操作码技术，那么最多还可设计出\_\_\_\_\_条单操作数指令。

- A.  $2^6$       B.  $(2^{4-m})'2^6 - 1$   
C.  $(2^{4-m})'2^6$       D.  $(2^{4-m})' (2^6 - 1)$

#### 试题分析

一条指令包括两部分：操作码和操作数。操作码用来说明指令的功能及操作性质；操作数用来指明操作码实施操作的对象。题中指令字长为16位，也就是说，一条指令最长是16位；对于双操作数指令而言，两个长度为6位的操作数共占去12位，剩余4位用作操作码，可设计出 $2^4=16$ 条这样的双操作数指令。现系统中已设计出m条双操作数指令，那么剩余的 $2^4-m$ 条可以用来设计单操作数的操作码。对于单操作数指令而言，它的操作码长度为 $16-6=10$ 位。由于题目要求采用扩展操作码技术，那么单操作数指令在原来的双操作数指令的4位操作码上可扩展 $10-4=6$ 位，因此，最多可设计出 $(2^{4-m})'2^6-1$ 条单操作数指令。减去1的原因是“存在无操作数指令”，至少留下一个用来扩展成无操作数指令。

例如，设计出了3条双操作数指令，操作码分别为0000、0001、0011，那么剩余的0100、0101、0110、...、1110、1111共13个可以作为单操作数指令操作码的一部分。由于能扩展6位，那么这13个中每一个都可以再扩展出 $2^6$ 个，如0100可扩展成 0100 000000、0100 000001、0100 000010、...、0100 111111。

#### 试题答案

B

### 例题11

#### 例题11（2007年5月试题1）

不属于计算机控制器中的部件。

- A. 指令寄存器 IR

- B. 程序计数器 PC
- C. 算术逻辑单元 ALU
- D. 程序状态字寄存器 PSW

#### 试题分析

控制器是分析和执行指令的部件，也是统一指挥和控制计算机各个部件按时序协调操作的部件。控制器的组成包含如下部分：① 程序计数器PC, ② 指令寄存器IR, ③ 指令译码器，④ 时序部件，⑤ 微操作控制信号形成部件PSW, ⑥ 中断机构.故C答案的算术逻辑单元ALU不属于控制器，是运算器。

#### 试题答案

C

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第 8 章：计算机组成与体系结构

作者：希赛教育软考学院 来源：希赛网 2014年02月10日

### 例题12

#### 例题12 ( 2007年5月试题2 )

在 CPU 与主存之间设置高速缓冲存储器 Cache,其目的是为了\_\_\_\_\_.

- A. 扩大主存的存储容量
- B. 提高 CPU 对主存的访问效率
- C. 既扩大主存容量又提高存取速度
- D. 提高外存储器的速度

#### 试题分析

由于CPU的速度比主存的读取速度快得多，为解决这种不匹配，在它们之间设置高速缓冲存储器Cache,将主存中的内容事先调入Cache中，CPU直接访问Cache的时间短得多，这样大大提高了CPU对主存的访问效率，也提高了整个计算机系统的效率。

#### 试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第 8 章：计算机组成与体系结构

作者：希赛教育软考学院 来源：希赛网 2014年02月10日

### 例题13

#### 例题13 ( 2007年5月试题3 )

下面的描述中，\_\_\_\_\_不是 RISC 设计应遵循的设计原则。

- A. 指令条数应少一些
- B. 寻址方式尽可能少
- C. 采用变长指令，功能复杂的指令长度长而简单指令长度短
- D. 设计尽可能多的通用寄存器

### 试题分析

RISC ( 精简指令系统计算机 ) 的设计原则有：

- 1、只用使用频度高的、以及最有用的指令，一般为几十条指令
- 2、指令格式简单化、规格化
- 3、每条指令在一个机器周期内完成
- 4、只有存数和取数指令访问存储器
- 5、以最简单有效的方式支持高级语言

很显然，C答案错误。

### 试题答案

C

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第 8 章：计算机组成与体系结构

作者：希赛教育软考学院 来源：希赛网 2014年02月10日

## 例题14

### 例题14 ( 2007年5月试题5 )

指令流水线将一条指令的执行过程分为四步，其中第 1、2 和 4 步的经过时间为 $\Delta t$ ，如图8-4所示。若该流水线顺序执行 50 条指令共用  $153\Delta t$ ，并且不考虑相关问题，则该流水线的瓶颈第 3 步的时间为 \_\_\_\_\_ $\Delta t$ 。

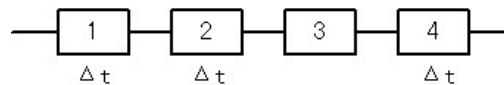


图8-4 流水线各段时间示意图

- A. 2
- B. 3
- C. 4
- D. 5

### 试题分析

采用流水线技术，执行周期取决于时间最长的步骤。本题的指令流水线将一条指令的执行过程分成四步，第1、2、4步的时间均为 $\Delta t$ ，而当描述第3步时用到了“瓶颈”一词，这说明第3步执行的时间最长，设为 $X\Delta t$ 。利用公式： $(1+1+X+1) + X * (50-1) = 153$ ，求得 $X=3$ 。所以第3步的时间为 $3\Delta t$ 。

### 试题答案：

B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

## 例题15

### 例题15 ( 2007年5月试题6 )

系统响应时间和作业吞吐量是衡量计算机系统性能的重要指标。对于一个持续处理业务的系统而言，其\_\_\_\_\_。

- A. 响应时间越短，作业吞吐量越小
- B. 响应时间越短，作业吞吐量越大
- C. 响应时间越长，作业吞吐量越大
- D. 响应时间不会影响作业吞吐量

#### 试题分析

系统响应时间是指用户发出完整请求到系统完成任务给出响应的时间间隔。作业吞吐量是指单位时间内系统完成的任务量。若一个给定系统持续地收到用户提交的任务请求，则系统的响应时间将对作业吞吐量造成一定影响。若每个任务的响应时间越短，则系统的空闲资源越多，整个系统在单位时间内完成的任务量将越大；反之，若响应时间越长，则系统的空闲资源越少，整个系统在单位时间内完成的任务量将越小。

#### 试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

## 例题16

### 例题16 ( 2007年11月试题1-2 )

在指令系统的各种寻址方式中，获取操作数最快的方式是（1）。若操作数的地址包含在指令中，则属于（2）方式。

- (1) A. 直接寻址 B. 立即寻址 C. 寄存器寻址 D. 间接寻址
- (2) A. 直接寻址 B. 立即寻址 C. 寄存器寻址 D. 间接寻址

#### 试题分析

此题考查的是考生对操作数几种基本寻址方式的理解。操作数寻址有以下方式：

##### 1、隐含寻址

在指令中不明显的给出而是隐含着操作数的地址。例如，单地址的指令格式，没有在地址字段中指明第二操作数地址，而是规定累加寄存器AC作为第二操作数地址，AC对单地址指令格式来说是隐含地址。

##### 2、立即寻址

指令的地址字段指出的不是操作数的地址，而是操作数本身。这种方式的特点是指令执行时间

很短，不需要访问内存取数。题目中所说的"操作数包含在指令中的寻址方式"就是立即寻址。

例如：单地址的移位指令格式为

OP(移位)	F	D
--------	---	---

这里D不是地址，而是一个操作数。F为标志位，当F=1,操作数进行右移；当F=0时，操作数进行左移。

### 3、直接寻址

直接寻址特点是：在指令格式的地址字段中直接指出操作数在内存的地址D.

采用直接寻址方式时，指令字中的形式地址D就是操作数的有效地址E,即E=D.因此通常把形式地址D又称为直接地址。此时，由寻址模式给予指示。如果用S表示操作数，那么直接寻址的逻辑表达式为  $S = (E) = (D)$

### 4、间接寻址

间接寻址的情况下，指令地址字段中的形式地址D不是操作数的真正地址，而是操作数地址的指示器，D单元的内容才是操作数的有效地址。

如果把直接寻址和间接寻址结合起来，指令有如下形式：

操作码	I	D
-----	---	---

寻址特征位I=0,表示直接寻址，这时有效地址E=D;I=1,表示间接寻址，这时有效地址E= (D)。

间接寻址方式是早期计算机中经常采用的方式，但由于两次访存，影响指令执行速度，现在已不大使用。

### 5、寄存器寻址方式和寄存器间接寻址方式

当操作数不放在内存中，而是放在CPU的通用寄存器中时，可采用寄存器寻址方式。此时指令中给出的操作数地址不是内存的地址单元号，而是通用寄存器的编号。这也就是题目中所说的"操作数在寄存器中的寻址方式".

寄存器间接寻址方式与寄存器寻址方式的区别在于：指令格式中的寄存器内容不是操作数，而是操作数的地址，该地址指明的操作数在内存中。这也就是题目中所说的"操作数的地址在寄存器中的寻址方式".

### 6、相对寻址方式

相对寻址是把程序计数器PC的内容加上指令格式中的形式地址D而形成操作数的有效地址。程序计数器的内容就是当前指令的地址。"相对"寻址，就是相对于当前的指令地址而言。

采用相对寻址方式的好处是程序员无须用指令的绝对地址编程，所编程序可以放在内存任何地方。

此时形式地址D通常称为偏移量，其值可正可负，相对于当前指令地址进行浮动。

### 7、基址寻址方式

基址寻址方式是将CPU中基址寄存器的内容加上指令格式中的形式地址而形成操作数的有效地址。它的优点是可以扩大寻址能力。同形式地址相比，基址寄存器的位数可以设置得很长，从而可以在较大的存储空间中寻址。

### 8、变址寻址方式

变址寻址方式与基址寻址方式计算有效地址的方法很相似，它把CPU中某个变址寄存器的内容与偏移量D相加来形成操作数有效地址。但使用变址寻址方式的目的不在于扩大寻址空间，而在于

实现程序块的规律性变化。

### 9、块寻址方式

块寻址方式经常用在输入输出指令中，以实现外存储器或外围设备同内存之间的数据块传送。

块寻址方式在内存中还可用于数据块搬家。块寻址时，通常在指令中指出数据块的起始地址（首地址）和数据块的长度（字数或字节数）。

如果数据块是变长的，可用三种方法指出它的长度：

- (1) 指令中划出字段指出长度；
- (2) 指令格式中指出数据块的首地址与末地址；
- (3) 由块结束字符指出数据块长度。

操作码	首地址	标志位	末地址
-----	-----	-----	-----

### 10、段寻址方式

微型机中采用了段寻址方式，例如它们可以给定一个20位的地址，从而有1M存储空间的直接寻址能力。为此将整个1M空间存储器以64K为单位划分成若干段。在寻址一个内存具体单元时，由一个基址再加上某些寄存器提供的16位偏移量来形成实际的20位物理地址。这个基地址就是CPU中的段寄存器。在形成20位物理地址时，段寄存器中的16位数会自动左移4位，然后以16位偏移量相加，即可形成所需的内存地址。这种寻址方式的实质还是基址寻址。

#### 试题答案

**B A**

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第 8 章：计算机组成与体系结构

作者：希赛教育软考学院 来源：希赛网 2014年02月10日

### 例题17

#### 例题17 ( 2007年11月试题3 )

系统响应时间和作业吞吐量是衡量计算机系统性能的重要指标。对于一个持续处理业务的系统而言\_\_\_\_\_，表明其性能越好。

- A. 响应时间越短，作业吞吐量越小
- B. 响应时间越短，作业吞吐量越大
- C. 响应时间越长，作业吞吐量越大
- D. 响应时间不会影响作业吞吐量

#### 试题分析

请参看例题15分析。

#### 试题答案

**B**

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

## 例题18

### 例题18 ( 2007年11月试题4-5 )

若每一条指令都可以分解为取指、分析和执行三步。已知取指时间 $t_{取指}=4\Delta t$ ,分析时间 $t_{分析}=3\Delta t$ ,执行时间 $t_{执行}=5\Delta t$ .如果按串行方式执行完100条指令需要  $(4+3+5)\Delta t$ .如果按照流水方式执行，执行完100条指令需要  $507\Delta t$ .

(4) A. 1190 B. 1195 C. 1200 D. 1205

(5) A. 504 B. 507 C. 508 D. 510

#### 试题分析

本题考查的是计算机系统指令流水线方面的基础知识。根据题意可以看到，在此流水线中按串行方式执行完100条指令要用 $1200\Delta t$ .采用流水方式执行，执行的总时间的关键取决于最长的执行时间，所以执行完100条的时间为： $4\Delta t+3\Delta t+5\Delta t+(100-1)\times5\Delta t=507\Delta t$ .

#### 试题答案

C B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

## 例题19

### 例题19 ( 2007年11月试题6 )

若内存地址区间为4000H~43FFH,每个存贮单元可存储16位二进制数，该内存区域用4片存储器芯片构成，则构成该内存所用的存储器芯片的容量是\_\_\_\_\_.

A.  $512\times16\text{bit}$  B.  $256\times8\text{bit}$  C.  $256\times16\text{bit}$  D.  $1024\times8\text{bit}$

#### 试题分析

本题考查内存容量的计算。给定起、止地址码的内存容量=终止地址-起始地址+1,所以：

$43FFH-4000H+1=400H$ ,十六进制的  $(400)_{16}=2^{10}$ ,组成内存储器的芯片数量级=内存储器的容量/单个芯片的容量。所以  $2^{10}/2^2=2^8$ .正确答案是C.

#### 试题答案

C

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

## 例题20

### 例题20 ( 2008年5月试题1 )

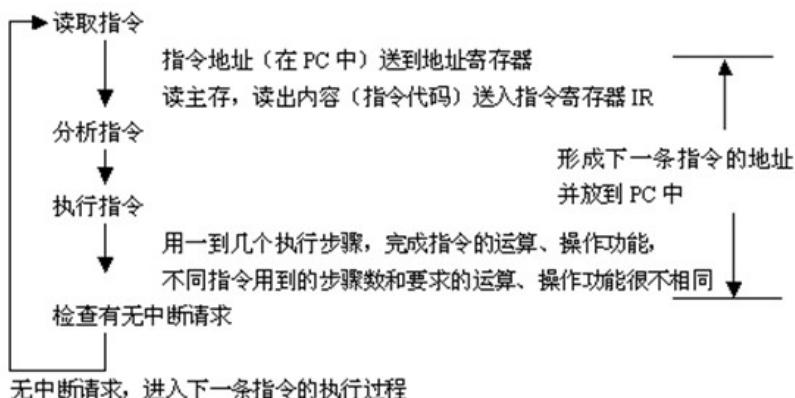
在计算机体系结构中，CPU 内部包括程序计数器 PC、存储器数据寄存器 MDR、指令寄存器IR 和存储器地址寄存器MAR 等。若CPU 要执行的指令为：MOV R0, #100 ( 即将数值100传送到寄存器R0中 )，则CPU 首先要完成的操作是\_\_\_\_\_。

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

#### 试题分析

本题考查计算机的硬件基础知识。

指令的执行步骤，一般要经过到内存读取指令，控制器分析指令，控制器按指令要求的具体操作功能，用一到几个执行步骤，“驱动”计算机相关部件完成指令的运算、操作功能，并在这一过程准备好下一条指令的地址到程序计数器PC中，至此本条指令的功能算是完成了，接下来检查有无中断请求，若无中断请求，则进入下一条指令的执行过程。这一执行过程可表示如图的形式。



所以CPU 要执行的指令为：MOV R0, #100, 则首先要把程序计数器 PC 的内容送到地址寄存器中。选择C答案。

#### 试题答案

C

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第 8 章：计算机组成与体系结构

作者：希赛教育软考学院 来源：希赛网 2014年02月10日

### 例题21

#### 例题21 ( 2008年5月试题2 )

现有四级指令流水线，分别完成取指、取数、运算、传送结果四步操作。若完成上述操作的时间依次为9ns、10ns、6ns、8ns,则流水线的操作周期应设计为\_\_\_\_\_ ns.

- A. 6 B. 8 C. 9 D. 10

#### 试题分析

本题考查计算机组成与体系结构的流水线问题。

流水线技术其实是通过并行硬件来提高系统性能的常用方法，其基本思想在冯·诺依曼第一台存储程序计算机中已经提出。流水线技术的基本原理实际上是一种任务分解的技术。把一件任务分解

成若干顺序执行的子任务，不同的子任务由不同的执行机构负责执行，而这些机构可以同时并行的工作。在任一时刻，任一任务只占用其中一个执行机构，这样就可以实现多个任务的重叠执行，以提高工作效率。流水线技术包括指令流水线和运算操作流水线。需要注意的是，对流水线技术而言，其对性能的提高程度取决于其执行顺序中最慢的一步。

在实际情况中，流水线各个阶段可能会相互影响，阻塞流水线，使其性能下降。影响流水线性能的主要因素有两个：执行转移指令和共享资源冲突。在实际处理中，为了使流水线能维持最大的吞吐率，同时确保流水线各段不会产生冲突，就需要对流水线进行很好的控制。一般采用预留表来预测冲突。预留表是从流水线设计直接推导出来的，表中列出的是流水线上各个部件操作的时间信息，每一行代表流水线中的一段，而每一列则代表一个时间步。

由流水线技术的基本特征可知，其平均时间取决于流水线最慢的操作，所以该流水线的操作周期为10ns.

### 试题答案

D

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第8章：计算机组成与体系结构

作者：希赛教育软考学院 来源：希赛网 2014年02月10日

## 例题22

### 例题22（2008年5月试题3）

内存按字节编址，地址从90000H 到CFFFFH,若用存储容量为16K×8bit的存储器芯片构成该内存，至少需要\_\_\_\_\_片。

- A. 2 B. 4 C. 8 D. 16

### 试题分析

本题考查计算机组成与体系结构中的内存编址，是常考的知识点。

存储地址空间是指对存储器编码（编码地址）的范围。所谓编码就是对每一个物理存储单元（一个字节）分配一个号码，通常叫作“编址”。内存按字节编址的方式求区间的大小，通常记住在“结束地址-起始地址”后还要加1H.所以题中地址从90000H 到CFFFFH的区间大小为：CFFFFH-90000H + 1H =40000H,将结果化成二进制为10000000000000000000,化成十进制为：4\*16<sup>4</sup>=218 .是以字节为单位，则大小为218 Byte = 28KB = 256 KB.用存储容量为16K×8bit的存储器芯片构成该内存，即256K÷16K = 16 ,所以至少需要16片。正确答案是D选项。

### 试题答案

D

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

## 例题23

### 例题23 ( 2008年5月试题4 )

CPU中的数据总线宽度会影响\_\_\_\_\_。

- A. 内存容量的大小
- B. 系统的运算速度
- C. 指令系统的指令数量
- D. 寄存器的宽度

#### 试题分析

本题考查计算机组成与体系结构的处理器的数据总线。

总线是一组物理导线，而非一根。根据总线上传送的信息不同，分为地址总线、数据总线和控制总线。

#### (1) 地址总线

地址总线传送地址信息。地址是识别信息存放位置的编号，主存储器的每个存储单元及I/O接口中不同的设备都有各自不同的地址。地址总线是CPU向主存储器和I/O接口传送地址信息的通道，它是自CPU向外传输的单向总线。

#### (2) 数据总线

数据总线传送系统中的数据或指令。数据总线是双向总线，一方面作为CPU向主存储器和I/O接口传送数据的通道。另一方面，是主存储器和I/O接口向CPU传送数据的通道，数据总线的宽度与CPU的字长有关。

#### (3) 控制总线

控制总线传送控制信号。控制总线是CPU向主存储器和I/O接口发出命令信号的通道，又是外界向CPU传送状态信息的通道。

数据总线负责整个系统的数据流量的大小，而数据总线宽度则决定了CPU与二级高速缓存、内存以及输入/输出设备之间一次数据传输的信息量。

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

所以说计算机系统的运算速度与CPU中的数据总线宽度有关，正确答案要选择B答案。

#### 试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

## 例题24

### 例题24 ( 2008年5月试题5 )

利用高速通信网络将多台高性能工作站或微型机互连构成机群系统，其系统结构形式属于\_\_\_\_\_计算机。

- A. 单指令流单数据流 ( SISD )
- B. 多指令流单数据流 ( MISD )
- C. 单指令流多数据流 ( SIMD )
- D. 多指令流多数据流 ( MIMD )

### 试题分析

本题考查计算机组成与体系结构的计算机分类，常考的知识点。

计算机系统的分类：Flynn分类、冯氏分类、Handler分类和Kuck分类；Flynn分类是根据不同指令流-数据流组织方式把计算机系统分成4类。

在系统性能的瓶颈部件上同时处于同样执行阶段的指令和数据的最大可能个数；

- I. 单指令流单数据流SISD--如单处理机
- II. 单指令流多数据流SIMD--如相联处理机
- III. 多指令流单数据流MISD--如流水线计算机
- IV. 多指令流多数据流MIMD--如多处理机

所以题中是利用高速通信网络将多台高性能工作站或微型机互连构成机群系统，事实上是采用了多处理机。

### 试题答案

D

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第8章：计算机组成与体系结构

作者：希赛教育软考学院 来源：希赛网 2014年02月10日

## 例题25

### 例题25 ( 2008年5月试题6 )

内存采用段式存储管理有许多优点，但“\_\_\_\_\_”不是其优点。

- A. 分段是信息的逻辑单位，用户不可见
- B. 各段程序的修改互不影响
- C. 地址变换速度快、内存碎片少
- D. 便于多道程序共享主存的某些段

### 试题分析

本题考查计算机的虚拟存储的段式存储系统。

在段式存储管理中，将程序的地址空间划分为若干个段 ( segment )，这样每个进程有一个二维的地址空间。在前面所介绍的动态分区分配方式中，系统为整个进程分配一个连续的内存空间。而在段式存储管理系统中，则为每个段分配一个连续的分区，而进程中的各个段可以不连续地存放内存的不同分区中。程序加载时，操作系统为所有段分配其所需内存，这些段不必连续，物理内存的管理采用动态分区的管理方法。在为某个段分配物理内存时，可以采用首先适配法、下次适配法、最佳适配法等方法。在回收某个段所占用的空间时，要注意将收回的空间与其相邻的空间合并。段式存储管理也需要硬件支持，实现逻辑地址到物理地址的映射。程序通过分段划分为多个模块，如代码段、数据段、共享段。这样做的优点是：可以分别编写和编译源程序的一个文件，并且可以针对不同类型的段采取不同的保护，也可以按段为单位来进行共享。总的来说，段式存储管理

的优点是：没有内碎片，外碎片可以通过内存紧缩来消除；便于实现内存共享。缺点与页式存储管理的缺点相同，进程必须全部装入内存。

段式和页式系统有许多相似之处。比如，两者都采用离散分配方式，且都通过地址映射机构来实现地址变换。但概念上两者也有很多区别，主要表现在：

页是信息的物理单位，分页是为了实现离散分配方式，以减少内存的外零头，提高内存的利用率。或者说，分页仅仅是由于系统管理的需要，而不是用户的需要。段是信息的逻辑单位，它含有一组其意义相对完整的信息。分段的目的是为了更好地满足用户的需要。

页的大小固定且由系统决定，把逻辑地址划分为页号和页内地址两部分，是由机器硬件实现的。段的长度不固定，且决定于用户所编写的程序，通常由编译系统在对源程序进行编译时根据信息的性质来划分。

页式系统地址空间是一维的，即单一的线性地址空间，程序员只需利用一个标识符，即可表示一个地址。分段的作业地址空间是二维的，程序员在标识一个地址时，既需给出段名，又需给出段内地址。

根据上面的分析，题中C选项不是段式存储系统的优点。

### 试题答案

C

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第8章：计算机组成与体系结构

作者：希赛教育软考学院 来源：希赛网 2014年02月10日

## 例题26

### 例题26 ( 2008年12月试题1 )

计算机内存一般分为静态数据区、代码区、栈区和堆区，若某指令的操作数之一采用立即数寻址方式，则该操作数位于\_\_\_\_\_。

- A. 静态数据区 B. 代码区 C. 栈区 D. 堆区

### 试题分析

本题是一个概念题，要求考生掌握两组概念。其一是关于内存的分区，即题目中的"静态数据区、代码区、栈区、堆区"各区的用途，其二是计算机寻址方式。

关于内存的分区，不同的编程语言分区方式略有不同，但基本都会包含题目所述的四大区域。在此我们以C++的分区方式来解释内存分区用途。在C++中，内存通常分成5个区，他们分别是栈区、堆区、全局/静态存储区、文字常量区和程序代码区。

1、栈区 ( stack )：由编译器自动分配释放（编译器在需要的时候分配，在不需要的时候自动清除），存放函数的参数值，局部变量的值等。其操作方式类似于数据结构中的栈。

2、堆区 ( heap )：一般由程序员分配释放，若程序员不释放，程序结束时可能由OS回收。它与数据结构中的堆是两回事，分配方式倒是类似于链表，一般用new来分配内存，用delete来释放内存。

3、全局区 ( 静态区 ) ( static )：全局变量和静态变量的存储是放在一块的，初始化的全局变

量和静态变量在一块区域，未初始化的全局变量和未初始化的静态变量在相邻的另一块区域，该区域是程序结束后由OS释放。

4、文字常量区：常量字符串就是放在这里的，程序结束后由OS释放。

5、程序代码区：存放函数体的二进制代码。

计算机的寻址方式有多种，如：直接寻址、间接寻址、立即数寻址、变址寻址、自动增量寻址、自动减量寻址、相对寻址等。其中的立即数寻址是在指令的地址码部分直接给出执行本条指令所需要的源操作数，它的主要优点是：节省了数据存储单元，指令的执行速度快；主要缺点是：只能用于源操作数的寻址，数据的长度不能太长，大量使用立即数寻址方式会使程序的通用性下降。

结合以上的概念可知，立即数寻址中的操作数是程序代码的一部分，所以应该存放在代码区。

### 试题答案

B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第8章：计算机组成与体系结构

作者：希赛教育软考学院 来源：希赛网 2014年02月10日

## 例题27

### 例题27（2008年12月试题2）

计算机在进行浮点数的相加（减）运算之前先进行对阶操作，若x的阶码大于y的阶码，则应将

- 
- A. x的阶码缩小至与y的阶码相同，且使x的尾数部分进行算术左移
  - B. x的阶码缩小至与y的阶码相同，且使x的尾数部分进行算术右移
  - C. y的阶码扩大至与x的阶码相同，且使y的尾数部分进行算术左移
  - D. y的阶码扩大至与x的阶码相同，且使y的尾数部分进行算术右移

### 试题分析

本题考查浮点数运算的基本过程，该过程分为如下三步：

第一步：对阶，规则是向大阶看齐。具体方法是：阶码小的那个尾数右移，移位的次数等于参加运算的两个数的阶码之差。

第二步：尾数相加。实际上进行的是定点数加法。

第三步：结果的后处理。包括如下3个方面：

- (1) 如果运算结果的尾数部分溢出，则需要向右规格化一位。具体方法是：尾数部分右移一位，阶码加1。
- (2) 如果运算结果的最高位为0，则需要向左规格化，并且要重复进行，直到尾数的最高位不为0。尾数部分每左移一位，阶码必须减1。
- (3) 进行舍入处理。

所以本题的正确答案为D。

### 试题答案

D

## 例题28

### 例题28 ( 2008年12月试题3 )

在CPU中，\_\_\_\_\_可用于传送和暂存用户数据，为ALU执行算术逻辑运算提供工作区。

- A. 程序计数器 B. 累加寄存器 C. 程序状态寄存器 D. 地址寄存器

#### 试题分析

本题考查CPU中常用寄存器的功能。下面将介绍这些寄存器的功能：

程序计数器 ( PC ) 是CPU内的一个寄存器，存储的是将要执行的下一条指令的地址，所以通常也叫做指令计数器。当执行指令时，CPU将自动修改PC的内容，以便使其保持的总是将要执行的下一条指令的地址。

累加寄存器AC通常简称为累加器，它是一个通用寄存器。其功能是：当运算器的算术逻辑单元ALU执行算术或逻辑运算时，为ALU提供一个工作区。累加寄存器暂时存放ALU运算的结果信息。显然，运算器中至少要有一个累加寄存器。目前CPU中的累加寄存器，多达16个，32个，甚至更多。当使用多个累加器时，就变成通用寄存器堆结构，其中任何一个可存放源操作数，也可存放结果操作数。在这种情况下，需要在指令格式中对寄存器号加以编址。

程序状态寄存器 ( Program Status Word,PSW )，又称标志寄存器。这是一个存放条件码标志，控制标志和系统标志的寄存器。

地址寄存器用来保存当前CPU所访问的内存单元的地址。由于在内存和CPU之间存在着操作速度上的差别，所以必须使用地址寄存器来保持地址信息，直到内存的读/写操作完成为止。当CPU和内存进行信息交换，即CPU向内存存/取数据时，或者CPU从内存中读出指令时，都要使用地址寄存器和数据缓冲寄存器。同样，如果我们把外围设备的设备地址作为像内存的地址单元那样来看待，那么，当CPU和外围设备交换信息时，我们同样使用地址寄存器和数据缓冲寄存器。

所以本题应选B.

#### 试题答案

B

## 例题29

### 例题29 ( 2008年12月试题4 )

下面关于在I/O设备与主机间交换数据的叙述，\_\_\_\_\_是错误的。

- A. 中断方式下，CPU需要执行程序来实现数据传送任务
- B. 中断方式和DMA方式下，CPU与I/O设备都可同步工作
- C. 中断方式和DMA方式中，快速I/O设备更适合采用中断方式传递数据
- D. 若同时接到DMA请求和中断请求，CPU优先响应DMA请求

### 试题分析

输入输出系统主要有五种方式与主机交换数据：程序控制方式、程序中断方式、DMA方式、通道方式、I/O处理机。而本题考查的正是其中的程序中断方式与DMA方式。

程序控制方式：CPU直接利用I/O指令编程，实现数据的I/O。CPU发出I/O命令，命令中包含了外设的地址信息和所要执行的操作，相应的I/O系统执行该命令并设置状态寄存器；CPU不停地（定期地）查询I/O系统以确定该操作是否完成。由程序主动查询外设，完成主机与外设间的数据传送，方法简单，硬件开销小。

程序中断方式：CPU利用中断方式完成数据的I/O，当I/O系统与外设交换数据时，CPU无需等待也不必去查询I/O的状态，当I/O系统完成了数据传输后则以中断信号通知CPU。CPU然后保存正在执行程序的现场，转入I/O中断服务程序完成与I/O系统的数据交换。然后返回原主程序继续执行。与程序控制方式相比，中断方式因为CPU无需等待而提高了效率。在系统中具有多个中断源的情况下，常用的处理方法有：多中断信号线法、中断软件查询法、维菊链法、总线仲裁法和中断向量表法。

DMA方式：使用DMA控制器（DMAC）来控制和管理数据传输。DMAC和CPU共享系统总线，并且具有独立访问存储器的能力。在进行DMA时，CPU放弃对系统总线的控制而由DMAC控制总线；由DMAC提供存储器地址及必须的读写控制信号，实现外设与存储器之间进行数据交换。

DMAC获取总线方式主要有三种，分别是暂停方式、周期窃取方式和共享方式。

DMA 传送方式的优先级高于程序中断，两者的区别主要表现在对CPU的干扰程度不同。中断请求不但使CPU停下来，而且要CPU执行中断服务程序为中断请求服务，这个请求包括了对断点和现场的处理以及CPU与外设的传送，所以CPU付出了很多的代价；DMA请求仅仅使CPU暂停一下，不需要对断点和现场的处理，并且是由DMA控制外设与主存之间的数据传送，无需CPU的干预，DMA只是借用了一点CPU的时间而已。

通道：通道是一种通过执行通道程序管理I/O操作的控制器，它使主机与I/O操作之间达到更高的并行程度。在具有通道处理机的系统中，当用户进程请求启动外设时，由操作系统根据I/O要求构造通道程序和通道状态字，将通道程序保存在主存中，并将通道程序的首地址放到通道地址字中，然后执行“启动I/O”指令。

输入输出处理机（IOP）：也称为外围处理机（PPU），它是一个专用处理机，也可以是一个通用的处理机，具有丰富的指令系统和完善的中断系统。专用于大型、高效的计算机系统处理外围设备的I/O，并利用共享存储器或其他共享手段与主机交换信息。从而使大型、高效的计算机系统更加高效地工作。与通道相比，IOP具有比较丰富的指令系统，结构接近于一般的处理机，有自己的局部存储器。

从以上的分析可以看出本题应选C。

### 试题答案

C

## 例题30

### 例题30 ( 2008年12月试题5 )

下面关于校验方法的叙述，\_\_\_\_\_是正确的。

- A. 采用奇偶校验可检测数据传输过程中出现一位数据错误的位置并加以纠正
- B. 采用海明校验可检测数据传输过程中出现一位数据错误的位置并加以纠正
- C. 采用海明校验，校验码的长度和位置可随机设定
- D. 采用CRC校验，需要将校验码分散开并插入数据的指定位置中

#### 试题分析

本题考查基本校验码，题目中提到的三种校验方式都是需要考生掌握的。

其先进度排名为：奇偶校验 < CRC校验 < 海明校验。

奇偶校验码是在源信息码的基础上添加一个校验位，使整个信息位呈奇性或偶性。这种校验码只能根据收到的信息奇偶性判断信息在传输中是否产生了1位错误，同时不能判断具体是哪一位出了错。

## 例题30

装置计算出CRC值并随数据一同发送给接收

装置，接收装置对收到的数据重新计算CRC并与收到的CRC相比较，若两个CRC值不同，则说明数据通讯出现错误。

海明校验是由Richard Hamming于1950年提出、目前还被广泛采用的一种很有效的校验方法，它只要增加少数几个校验位，可以发现"≤码距-1"位的错误、可以纠正"≤码距/2"位的错误，因此如果要能够纠正n位错误，则所需最小的码距应该是"2n+1"。如要纠正1位错误，则最小码距为3。

有了以上的知识基础，问题就非常容易解决了，接下来逐个分析备选答案中存在的概念错误：

- A、"采用奇偶校验可检测数据传输过程中出现一位数据错误的位置并加以纠正" 奇偶校验码只能检测一位的错误，不能确定错误的位置，更无法对错误进行纠正，描述错误。
- B、"采用海明校验可检测数据传输过程中出现一位数据错误的位置并加以纠正" 海明码能发现错误位置，并予以修改，描述正确。
- C、"采用海明校验，校验码的长度和位置可随机设定"，海明码的编码过程有严格要求，对于信息位与校验位的放置也是有约定的，不能随机设定，描述错误。
- D、"采用CRC校验，需要将校验码分散开并插入数据的指定位置中"，CRC码的校验位都是置于编码的最后部分（最右端），描述错误。

故正确答案为：B.

#### 试题答案

B

**例题31****例题31 ( 2008年12月试题6 )**

Cache用于存放主存数据的部分拷贝，主存单元地址与Cache单元地址之间的转换工作由\_\_\_\_\_完成。

- A. 硬件 B. 软件 C. 用户 D. 程序员

**试题分析**

概念题，主存单元地址与Cache单元地址之间的转换工作是由硬件完成的。

**试题答案**

**A**

**例题1****9.2 试题精解****例题1 ( 2005年5月试题7 )**

三个可靠度R均为0.8的部件串联构成一个系统，如图9-1所示。



**图9-1 串联系统**

则该系统的可靠度为\_\_\_\_\_。

- A.0.240 B.0.512 C.0.800 D.0.992

**试题分析**

此题是一个串联系统可靠性计算的试题，可靠性 $R=0.8 \times 0.8 \times 0.8 = 0.512$ 。

**试题答案**

**B**