

计算机组成原理复习要点

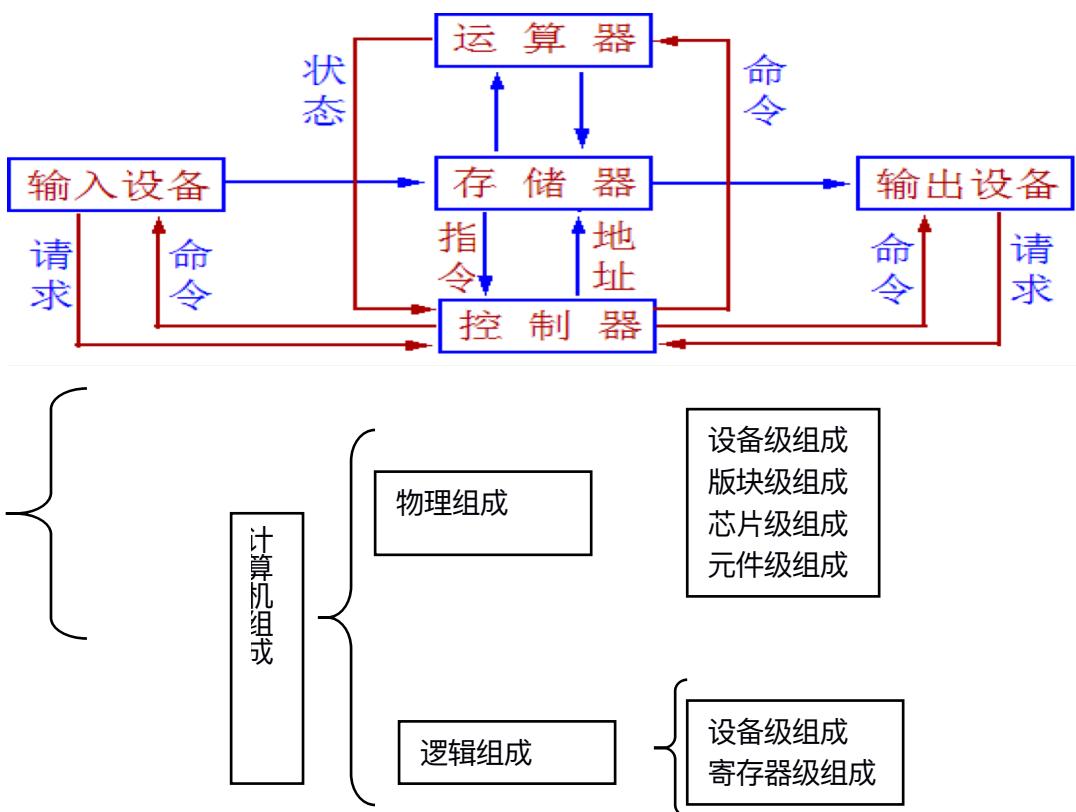
一、题型分布

选择题 20 分；填空题 30 分；判断题 10 分；计算题 20/25 分；简答题 20/15 分

二、每章重点内容

第一章 概述

1、什么是计算机组成



2、诺依曼体系结构计算机的特点

(1) 硬件由五大部份组成 (运算器、控制器、存储器、输入设备、输出设备)。

(2) 软件以 2#表示。

(3) 采用存储程序

所有的程序预先存放在存储器中,此为计算机高速自动的基础;

存储器采用一维线性结构;

指令采用串行执行方式。

控制流 (指令流) 驱动方式;

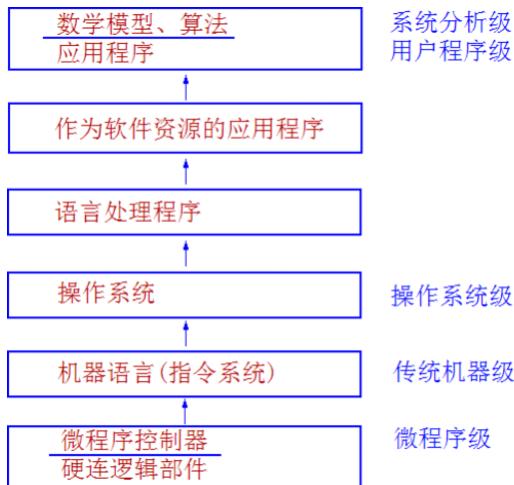
(4) 非诺依曼体系结构计算机

数据流计算机

多核(芯)处理机的计算机

3、计算机系统的层次结构

(1) 从软、硬件组成角度划分层次结构



(2) 从语言功能角度划分的层次结构

虚拟机：通过软件配置扩充机器功能后，所形成的计算机，实际硬件并不具备相应语言的功能。

第二章 数据表示

1、各种码制间的转换及定点小数和定点整数的表示范围

(1) 原码：

计算规则：最高位表示符号位；其余有效值部分以 2# 的绝对值表示。如：

$$(+0.1011) \text{ 原} = 0.1011; \quad (-0.1001) \text{ 原} = 1.1001$$

$$(+1011) \text{ 原} = 01011; \quad (-1001) \text{ 原} = 11001$$

注意：在书面表示中须写出小数点，实际上在计算机中并不表示和存储小数点。

原码的数学定义

若定点小数原码序列为 $X_0.X_1X_2...X_n$ 共 $n+1$ 位数，则：

$$X \text{ 原} = X \quad \text{当 } 1 > X \geq 0$$

$$X \text{ 原} = 1-X = 1+|x| \quad \text{当 } 0 \geq X > -1$$

若定点整数原码序列为 $X_0X_1X_2...X_n$ 共 $n+1$ 位数，则：

$$X \text{ 原} = X \quad \text{当 } 2n > X \geq 0$$

$$X \text{ 原} = 2^n-X = 2^n+|x| \quad \text{当 } 0 \geq X > -2^n$$

说明：

在各种码制（包括原码）的表示中需注意表示位数的约定，即不同的位数表示结果不同，如：

以 5 位表示，则 $(-0.1011) \text{ 原} = 1.1011$

以 8 位表示，则 $(-0.1011) \text{ 原} = 1.1011000$

0 的原码有二种表示方式：

小数： $(+0.0000) \text{ 原} = 0.0000, (-0.0000) \text{ 原} = 1.0000$

整数： $(+00000) \text{ 原} = 00000, (-00000) \text{ 原} = 10000$

符号位不是数值的一部分，不能直接参与运算，需单独处理。

约定数据位数的目的是约定数据的表示范围，即：

小数： $-1 < X < 1$

整数： $-2^n < X < 2^n$

(2) 反码：

计算规则：正数的反码与原码同；负数的反码是原码除最高位（符号位）外，各位求反。如：

正数：(+0.1011) 原= (+0.1011) 反=0.1011;

负数：(-0.1001)原=1.1001，则(-0.1001)反=1.0110

反码的数学定义

若定点小数反码序列为 $X_0.X_1X_2...X_n$ 共 $n+1$ 位数，则：

$$X_{\text{反}} = X \quad \text{当 } 1 > X \geq 0$$

$$X_{\text{反}} = (2-2-n) + X \quad \text{当 } 0 \geq X > -1$$

若定点整数反码序列为 $X_0X_1X_2...X_n$ 共 $n+1$ 位数，则：

$$X_{\text{反}} = X \quad \text{当 } 2n > X \geq 0$$

$$X_{\text{反}} = (2n-1) + X \quad \text{当 } 0 \geq X > -2n$$

(3) 补码：

计算规则：正数的补码与原码同；负数的补码是反码的最低加 1。如：

正数：

(+0.1011)原=(+0.1011)反=(+0.1011)补=0.1011;

负数：

(-0.1001)原=1.1001

(-0.1001)反=1.0110

(-0.1001)补=1.0111

数学定义

$$(X)_{\text{补}} = M + X \pmod{M}$$

其中：M 表示模，即容器的最大容量。

若定点小数补码序列为 $X_0.X_1X_2...X_n$ 共 $n+1$ 位数，则 $M=2$ ；

若定点整数补码序列为 $X_0X_1X_2...X_n$ 共 $n+1$ 位数，则 $M=2n+1$

2、为什么计算机中数值类型的数据以补码表示

补码的符号位是数值的一部分，可以参与运算。

0 的补码表示具有唯一性。

补码的表示范围比原码、反码大。

3、常见寻址方式的特点

(1) 寻址方式：获得指令或操作数的方式。

(2) 指令寻址：由程序计数器提供即将要执行的指令的地址。

(3) 操作数寻址：与具体的寻址方式有关。操作数寻址方式应说明是源操作数还是目标操作数的寻址方式。

4、采用多种寻址方式的目的（缩短指令长度，扩大寻址空间，提高编程灵活性）

缩短指令长度，扩大寻址空间，提高编程的灵活性。

5、如何减少指令中地址数的方法

采用隐地址（隐含约定）可以简化指令地址结构，即减少指令中的显地址数。

6、外设的编址方式（在任何一种方式每个外设都有一个独立的地址）

(1) I/O 与主存统一编址，即 I/O 是看作是主存的延伸。

(2) I/O 与主存单独编址：

I/O 编址到设备级，即一个 I/O 只有一个地址。

I/O 编址到寄存器级，即一个 I/O 有多个地址。

7、指令系统优化的趋势（CISC、RISC）

(1) CISC(复杂指令系统计算机)

从编程角度出发，希望指令系统中包含的指令尽可能多，每条指令中的操作信息尽可能多。该类指令系统一般包含 300-500 指令。为提高机器效率，采用了向量化、超标量、超长指令字等技术。

(2) 指令系统的发展趋势

早期：面向用户编程，采用 CISC 技术

现代：面向系统、向高级语言靠近，采用 RISC 技术

(3) 实际上 CISC 和 RISC 均是当前的发展(优化)趋势

第三章 存储器

1、存储器的按工作原理和存取方式的分类

(1) 物理原理分类：A、磁芯 B、半导体存储器 C、磁表面存储器 D、光盘存储器 E、其它存储器

(2) 存取方式的划分：A、随机存取存储器 (RAM) B、只读存储器 (ROM) C、顺序存取存储器 (SAM) D 直接存取存储器 (DAM)

2、存储器的三级层次结构及各层次的功能

(1) 主存：基本要求：随机访问、工作速度快、具有一定容量；

功能：存放当前执行的指令和数据。

(2) 外存：基本要求：容量大、成本低、一定的速度

功能：长期保存数据；作为主存的外援存储器。

外存也可采用多级存储结构。

(3) cache：基本要求：速度足够快、一定容量

功能：CPU 与主存的缓冲，匹配主存与 CPU 的速度。

内容：是当前主存中最活跃数据的副本。

内容形成的依据：

程序局部性原理：时间和空间局部性。

3、静态与动态存储器间的区别、动态存储器为什么还需要刷新及刷新有分类

(1) 根据信息表示方式分为：

动态存储器(DRAM)：以电容中的电荷表示信息，需动态刷新；

静态存储器(SRAM)：以双稳态信息。

(2) 需动态刷新：因为动态存储器是依靠电容上的存储电荷暂存信息，而电容上存储的电荷会逐渐减变弱所以需要刷新。

(3) 刷新的分类：A、集中刷新 B、分散刷新 C、异步刷新。

4、校验码：奇偶、循环校验码 (CRC) 计算

(1) 奇/偶校验：

奇/偶校验：使校验码中“1”的个数和为奇/偶数，主要用于主存校验。

例：有效信息：01101011，则

奇校验码：011010110

偶校验码：011010111

(2) 循环校验码

A、编码原理：

现假设有：

有效信息：M ； 除数 G (生成多项式)

有：

$$M/G = Q + R/G$$

此时，可选择 R 作为校验位，则 MR 即为校验码。

B、校验原理：

$$(M-R) \text{ } /G=Q+0/G$$

说明：以接收到的校验码除以约定的除数，若余数为 0，则可认为接收到的数据是正确的。

例：有效信息 1101，生成多项式样 1011，求循环校验码

解：

有效信息 1101($k=4$)，即 $M(x)=x^3+x^2+x^0$

生成多项式 1011($r+1=4$, 即 $r=3$), 即 $G(x)=x^3+x^1+x^0$

$M(x) \cdot x^3 = x^6+x^5+x^3$, 即 1101000 (对 1101 左移三位)

$M(x) \cdot x^3 / G(x) = 1101000 / 1011 = 1111 + 001 / 1011$

即 1010 的 CRC 是：1101001

计算过程	校验过程
$\begin{array}{r} 1111 \\ 1011 \longdiv{1101000} \\ \underline{\oplus 1011} \\ 1100 \\ \underline{\oplus 1011} \\ 1110 \\ \underline{\oplus 1011} \\ 1010 \\ \underline{\oplus 1011} \\ 001 \end{array}$	$\begin{array}{r} 1111 \\ 1011 \longdiv{1101001} \\ \underline{\oplus 1011} \\ 1100 \\ \underline{\oplus 1011} \\ 1110 \\ \underline{\oplus 1011} \\ 1011 \\ \underline{\oplus 1011} \\ 000 \end{array}$
余数为 001	余数为 001

循环校验码的来源

余数与出错序号间处理存在对应模式，该模式只与只与码制和生成多项式有关，与具体的码字无关。

生成多项式满足的条件：

任一位发生错误都应使余数不为 0；

不同的位发生的错误余数应不同。

用的生成多项式：

CCITT: $G(x)=x^{16}+x^{15}+x^2+1$

IEEE: $G(x)=x^{16}+x^{12}+x^5+1$

5、存储器的扩展

(1) 位扩展：

例： $2K \times 4$ 芯片组成 $2K \times 8$

特点：

(1) 片选信号连接在一起，二个芯片分别提供高低位的数据；

(2) 芯片的地址线直接与 AB 按位连接。

(2) 字扩展

例： $2K \times 4$ 芯片组成 $4K \times 4$

特点：

AB 高位地址通过译码形成芯片的片选信号；

AB 低位地址通过译码连接芯片的低位地址；

(3) 综合扩展

例： $4K \times 4$ 芯片组成 $16K \times 8$

6、数据传输率的计算

$$R = \frac{1}{cpu\text{主频}} \text{ (单位 bps)}$$

7、提高存储性能（速度、容量）的措施

A、双端口存储器,B、并行主存系统 C、高速缓存 D、虚拟存储 E、相联存储技术等。

8、高速缓存的功能及替换算法

- (1) 高速缓存的功能：提供的是 cpu 与内存的一个缓存。
- (2) 替换算法：1 先进先出算法(FIFO)2 近期最少使用算法(LRU)

$$p\text{ 命中率} = \frac{\text{访问cache次数}}{\text{访问总次数} (\text{访问内存数} + \text{访问cache次数})}$$

9、Cache 与内存在直接映像方式中怎样将内存地址转换为 Cache 地址

A、直接映像 B、全相联映像 C、组相联映像。

10、虚拟存储器的分类

A、页式虚拟存储器 B、段式虚拟存储器 C、段页式虚拟存储器。

第四章 CPU

1、为什么会产生溢出、及溢出的解决方法、正负溢出的概念

- (1) 产生溢出的原因：需表示的数据或运算结果超出了正常表示范围
- (2) 溢出的解决方法：多符号位；
- (3) 正溢出：两个正数相加而绝对值超出允许的表示范围；
- (4) 负溢出：两个负数相加而绝对值超出允许的表示范围。

2、补码加减法的依据

$X\text{ 补}+Y\text{ 补}=(X+Y)\text{ 补}$ 和 $X\text{ 补}-Y\text{ 补}=X\text{ 补}+(-Y)\text{ 补}$ 。

3、串行和并行加法的原理

串行加法原理如下：

$$C_1 = G_1 + P_1 C_0 \quad ; \text{ 其中 } C_0=0$$

$$C_2 = G_2 + P_2 C_1$$

....

$$C_n = G_n + P_n C_{n-1}$$

$$\sum_i A_i \oplus B_i \oplus C_{i-1}$$

并行加法原理如下：

$$C_1 = G_1 + P_1 C_0$$

$$C_2 = G_2 + P_2 C_1 = G_2 + P_2 G_1 + P_2 P_1 C_0$$

$$C_3 = G_3 + P_3 C_2 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 C_0$$

$$C_4 = G_4 + P_4 C_3$$

$$= G_4 + P_4 G_3 + P_4 P_3 G_2 + P_4 P_3 P_2 G_1 + P_4 P_3 P_2 P_1 C_0$$

....

$$\text{而 } \sum_i A_i \oplus B_i \oplus C_{i-1}$$

4、一位原码乘法的计算及运算特点

(1)数学原理：

两个原码数相乘,其乘积的符号为相乘两数符号的异或值,数值等于两数绝对值之积。

假设 $[X]_{\text{原}}=X_0.X_1X_2..X_n$,

$[Y]_{\text{原}}=Y_0.Y_1Y_2..Y_n$, 则有:

$$[X \cdot Y]_{\text{原}} = (X_0 \oplus Y_0) \cdot [(X_1X_2..X_n) \cdot (Y_1Y_2..Y_n)]$$

(2)算法：

假设 $X=0.X_1X_2..X_n$, $Y=0.Y_1Y_2..Y_n$, 即均为正纯小数

$$X \cdot Y = X \cdot 0.Y_1Y_2..Y_n$$

$$= X \cdot (2^{-1}Y_1 + 2^{-2}Y_2 + \dots + 2^{-n+1}Y_n - 1 + 2^{-n}Y_n)$$

$$= X \cdot (2^{-n}Y_n + 2^{-n+1}Y_n - 1 + \dots + 2^{-2}Y_2 + 2^{-1}Y_1)$$

$$= ((0+Y_nX)2^{-1} + (Y_n-1X)2^{-2} + \dots + (Y_2X)2^{-1} + (Y_1X)2^{-1})$$

根据上述计算过程, 可得算法如下:

$$A_0=0$$

$$A_1= (A_0+Y_nX)2^{-1}$$

$$A_2= (A_1+Y_{n-1}X)2^{-1}$$

...

$$A_{n-1}= (A_{n-2}+Y_2X)2^{-1}$$

$$A_n= (A_{n-1}+Y_1X)2^{-1}$$

积 $X \cdot Y = A_n$

乐山师范学院 Leshan Teachers College

例1: $X=0.1101, Y=-0.1011, \text{求}(Y \cdot Y)$

解: 累加和 $A=0.0000$ (防止溢出采用双符号位)
被乘数 $B=|X|=0.1101$ (与 A 同, 采用双符号位); 乘数 $C=|Y|=.1011$

步数	条件	操作	A	C	判断位Cn
			00.0000	.1011	C4
(1)	$C_4=1$	$+B$	$+ 00.1101$		
			<u>00.1101</u>		
(2)	$C_3=1$	$+B$	$+ 00.1101$	1.101	C3
			<u>01.0011</u>		
(3)	$C_2=0$	\rightarrow	00.1001	11.10	C2
(4)	$C_1=1$	$+B$	$+ 00.1101$	111.1	C1
			<u>01.0001</u>		
		\rightarrow	00.1000	1111.	



例2: $X=-0.1011, Y=-0.0110, \text{求}(Y \cdot Y)_{\text{原}}$

解: 累加和 $A=0.0000$, 被乘数 $B=|X|=0.1011$, 乘数 $C=|Y|=.0110$

步数	条件	操作	A	C	判断位Cn
			00.0000	.0101	C4

$$(1) C_4=0 \rightarrow 00.0000 \quad 0.0101 \quad C3$$

$$(2) C_3=1 \quad +B \quad + 00.1011$$

$$\underline{00.1011}$$

$$(3) C_2=1 \quad +B \quad + 00.1011 \quad 10.01 \quad C2$$

$$\underline{01.0000}$$

$$(4) C_1=0 \quad \rightarrow \quad 00.0100 \quad 010.0 \quad C1$$

符号位 $=X_f \oplus Y_f = 1 \oplus 0 = 0$

乘积的绝对值存放在AC中

因此, $(Y \cdot Y)_{\text{原}}=0.0100 \quad 0010$

(3) 运算特点

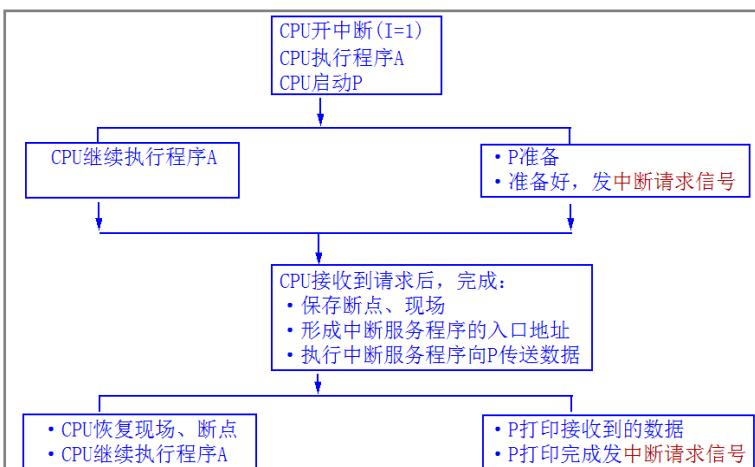
符号位和绝对值分别独立运算。

5、主机与外设间的连接方式

- (1) 辐射型
- (2) 总线型
- (3) 通道型

6、数据传送方式

- (1) 以打印机为例说明中断方式数据的传输过程



(2) DMA 方式的特点及应用

DMA 方式的特点：传送速率快，操作简单；应用：高速外部设备与主存储器之间的简单批量数据传送。

中断的响应过程、区别中断处理与中断服务

7、中断的响应过程、区别中断处理与中断服务

- (1) 中断的响应过程

① 置位中断优先级有效触发器，即关闭同级和低级中断：

② 调用入口地址，断点入栈，相当于 LCALL 指令；
③ 进入中断服务程序。

④ 中断处理就是执行中断服务程序，从中断入口地址开始执行，直到返回指令(RET)为止。此过程一般包括三部分内容，一是保护现场，二是处理中断源的请求，三是恢复现场。

⑤ 中断服务是要完成处理的事务，用户根据需要编写中断服务程序，程序中要注意将主程序中需要保护的寄存器内容进行保护

8、控制器的功能

CPU 内部控制功能的寄存器及相应的功能

硬件系统时序层次的划分及各层次的含义

控制器的分类及各自的优缺点

指令流程 (MOV 和运算类双操作数指令)

微程序控制器的基本思想

9、CPU 内部控制功能的寄存器及相应的功能

- (1) PC：程序计数器，用来指示指令在存储器中的存放位置。
- (2) IR：指令寄存器，用来存放当前正在执行的指令，它的输出包括操作信息、地址信息等。
- (3) PSW：程序状态寄存器，用来记录现行程序的运行状态和指示程序的工作

方式。

10、硬件系统时序层次的划分及各层次次的含义

- (1) 指令周期：执行一条指令所需的时间，一般由若干个机器周期（工作周期）组成，是从取指令、分指令到执行指令完所需的全部时间。
- (2) 工作周期：完成一个阶段性的任务所需时间。
- (3) 时钟周期：完成一个基本操作所需时间。
- (4) 定时脉冲：启停控制逻辑的时钟脉冲。（工作脉冲：决定一个周期的长短）

11、控制器的分类及各自的优缺点

- (1) 组合逻辑控制

A、优点：思路简单、可用于实现任一指令系统。

B、缺点：

*控制器的核心零乱、繁琐，设计效率低，检查调试困难。

*不易扩展和修改。

- (2) 微程序控制

A、优点：

*用规整的存储逻辑结构代替硬连逻辑，有利于设计自动化。

*易于修改与扩展，灵活，通用性强。

*适用性作素列机的控制器。

*可靠性高，易于诊断与维护。

B、缺点：

*速度慢，效率低。

*由于增加了相关的微程序控制部件，成本较高。

12、指令流程（MOV 和运算类双操作数指令）

- (1) MOV

三、MOV指令

1、FT

FT0: M \rightarrow IR 、 PC+1 \rightarrow PC

2、ST:

由源操作数的寻址方式确定 ST

指令流程：若源数在内存中，
则取到C；若源数在R中，则省略。



节拍	R	(R)	n(立即)	@(R) +	X(R)
ST0		R \rightarrow MAR	PC \rightarrow MAR	R \rightarrow MAR	PC \rightarrow MAR
ST1		M \rightarrow MDR \rightarrow C			
ST2			PC+1 \rightarrow PC	R+1 \rightarrow R	PC+1 \rightarrow PC
ST3				C \rightarrow MAR	R+C \rightarrow MAR
ST4	↓			M \rightarrow C	M \rightarrow MDR \rightarrow C

3、DT

由目的操作数的寻址方式确定 DT 的指令流程：若目的数在内存中，则将目的数的地址送到 MAR；若目的数在 R 中，则省略。

节拍	R	(R)	n(立即)	@(R) +	X(R)
DT0		R→MAR	无	R→MAR	PC→MAR
DT1				R+1→R	PC+1→PC
DT2				M→MDR→MAR	M→MDR→D
DT3	↓				R+D→MAR

4、ET

由源操作数、目的操作数的寻址方式及操作码共同确定 ET 的指令流程。

源数可在 R 中(SR)或内存中(SR);

目的数可在 R 中(DR)或内存中(DR)。

由源数和目的数的位置可以有下列四种类型：

节拍	SR • DR	SR • \bar{DR}	$\bar{SR} • DR$	$\bar{SR} • \bar{DR}$
ET0	R源→R目的	R→MDR	C→R	C→MDR
ET1	PC→MAR	MDR→M	PC→MAR	MDR→M
ET2		PC→MAR		PC→MAR

(2) 双操作数指令

包括：ADD、SUB、OR、AND、EOR。

双操作数指令的 FT 和 ST 的流程与 MOV 指令完全相同。下面只描述 DT 及 ET 的指令流程。

1、DT

由目的操作数的寻址方式确定 DT 的指令流程：若目的数在内存中，则将目的数送到 D；若目的数在 R 中，则省略。

节拍	R	(R)	n	@(R) +	X(R)
DT0		R→MAR	无	R→MAR	PC→MAR
DT1		M→MDR→D		R+1→R	PC+1→PC
DT2				M→MDR→D	M→MDR→D
DT3				D→MAR	R+D→MAR
DT4	↓			M→MDR→D	M→MDR→D

2、ET

由源操作数、目的操作数的寻址方式及操作码共同确定 ET 的指令流程。

源数可在 R 中(SR)或内存中(SR);

目的数可在 R 中(DR)或内存中(DR)。

由源数和目的数的位置可以有下列四种类型。

其中：Rs 表示源寄存器； Rd 表示目的寄存器。

节拍	SR • DR	SR • \bar{DR}	$\bar{SR} • DR$	$\bar{SR} • \bar{DR}$
ET0	Rs OP Rd→Rd	R OP D→MDR	C OP R→R	C OP D→MDR
ET1	PC→MAR	MDR→M	PC→MAR	MDR→M
ET2		PC→MAR		PC→MAR

13、微程序控制器的基本思想

(1) 机器指令由微程序解释；微程序由微指令组成，每条微指令中可包含多个微命令；微命令控制实现微操作。

(2) 微指令以代码(微码)存储在 ROM 中，该 ROM 称为控制存储器 (CM)。

9、I/O 系统

1、总线的分类及接口的分类

(1) 总线的分类

- A、按据传送格式分类：串行总线；并行总线
- B、按时序控制方式分类：同步总线；异步总线

(2) 接口的分类

- A、按数据传送格式划分

并行接口：接口与系统总线及 I/O 间均以并行方式传送数据。

串行接口：接口与 I/O 间以串行方式，而与系统总线间以并行方式传送数据。

- B、按时序划分

同步接口：与同步总线连接的接口，接口与系统总线间的数据传送由统一的时序信号(由 CPU 或专门的系统总线时序信号)控制。

异步接口：与异步总线连接的接口，接口与系统总线间的传送采用异步应答的工作方式。

- C、按信息的传送控制方式划分

中断接口、DMA 接口、磁盘存储接口等。

***I/O 接口的基本功能

(1)寻址：将地址信息译码为 I/O 或接口中的寄存器的选中信号。

(2)数据传送与缓冲(速度匹配)

(3)数据格式变换、电平变换等预处理

(4)控制逻辑：

接口对主机发送的命令字进行解释，并将产生的操作命令发送给 I/O；

将 I/O 及接口的状态信息送回 CPU。如在中断接口中有中断请求信号产生、中断屏蔽、优先排队等部件。

2、总线的概念及分时共享的含义、总线的组成

总线 (Bus) 是计算机各种功能部件之间传送信息的公共通信干线，它是由导线组成的传输线束，按照计算机所传输的信息种类，计算机的总线可以划分为数据总线、地址总线和控制总线，分别用来传输数据、数据地址和控制信号。总线是一种内部结构，它是 cpu、内存、输入、输出设备传递信息的公用通道，主机的各个部件通过总线相连接，外部设备通过相应的接口电路再与总线相连接，从而形成了计算机硬件系统。在计算机系统中，各个部件之间传送信息的公共通路叫总线，微型计算机是以总线结构来连接各个功能部件的。

3、PCI 总线的仲裁方式

当多个设备竞争总线时，由总线仲裁器进行仲裁，在微机中就是南北桥芯片组。
仲裁方式分集中和竞争二种方式。

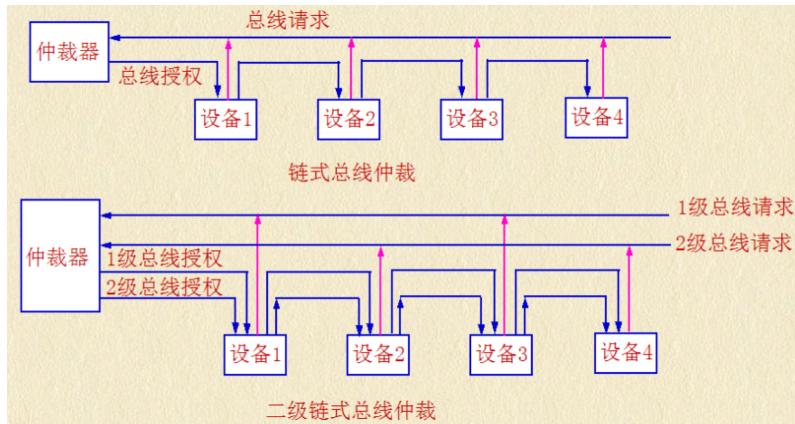
(1) 集中式仲裁：

分配原则是：优先级高的设备可以剥夺优先级低的设备的总线使用权。

当仲裁器接收到总线请求时，就发出总线授权信号。

设备的优先级由设备和仲裁器的逻辑距离决定。

当设备较多时，可采用分级仲裁方式。

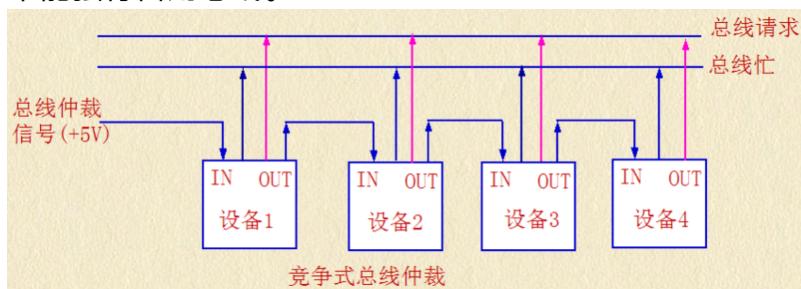


(2) 竞争式仲裁

分配原则：

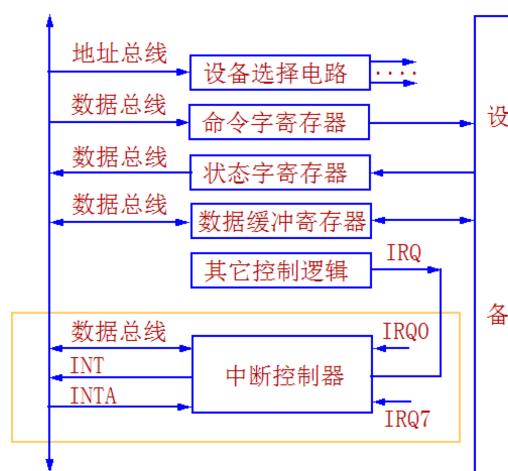
优先级不同的设备同时申请总线，则分配给高优先级的设备；

先来先占用：谁先抢占总线，只要该设备没有释放总线，则其它优先级高的设备不能强行占用总线。



4、中断接口的模型及工作过程

中断接口的模型



(1) 设备选择电路

是一个译码器，用于选择接口中的某一个寄存器。

(2) 命令字寄存器

用于接收 CPU 发送的命令字，一般用于初始化接口，如数据的输入/出方向、工作方式(R 或 W)等。

(3) 状态字寄存器

用以记录、反映设备与接口的运行状态，作为 CPU 执行 I/O 程序的依据。

(4) 数据缓冲寄存器

作为主机与 I/O 间数据传送的缓冲。

其容量称为缓冲深度。若对缓冲深度要求较高，则可采用半导体存储器作为缓冲区。

(5) 其它控制逻辑

接口不同，则该逻辑不同，属接口中不规整的部分，一般有如下部分：

A、中断请求 IRQ 的产生

B、与主机间的应答逻辑

C、控制时序，包括振荡电路、分频电路。

D、面向设备的某些特殊逻辑。如对机电性的设备所需的电机的启动、停止、正转、反转、加速，数据格式的转换，电平信号的转换等

E、智能控制器。功能复杂的接口，常使用通用的微处理器、单片机或专用控制器等芯片，与半导体存储器构成可编程的控制器。

(6) 中断控制器

现通常采用 8259A 芯片。

****模型接口的抽象工作过程：

(1) 初始化接口与中断控制器：CPU 调用程序或系统初始化时完成

(2) 启动外设：通过专门的启动信号或命令字，使接口状态为 B=1、D=0。

(3) 设备向中断控制器提出中断请求：设备准备好或完成一次操作(数据传送)，使接口状态为：B=0、D=1，据此形成中断请求信号 IRQi。

(4) 中断控制器向 CPU 提出中断请求：IRQi 送中断控制器 8259A，经屏蔽优先，向 CPU 发出公共请求 INT，同时形成中断类型码。

(5) CPU 响应：CPU 经中断判优后向 8259A 发回响应信号 INTA，CPU 从 DB 取回中断类型码。

(6) CPU 在中断周期 IT 执行中断隐指令操作，转入中断服务程序。

第六章 I/O 设备

1、键盘对按键识别方式的分类

软件扫描、硬件扫描

2、显存容量的计算

显存容量应该能保证一帧图像的存储。下面以例说明：

例：显示器在字符方式下：分辨率 25 行×80 列，颜色 256 种。显示字符集有 256 个字符；在图形方式下分辨率 1024×1024，颜色 216 种

解：(1)字符方式下：

256 个字符集中的每个字符需 8 位(1B)表示，256 种颜色需 8 位(1B)表示，则：

基本缓存： $25 \times 80 \times 8 / 8 = 2000B$, 2KB

属性缓存： $2KB \times 1 = 2KB$

显存容量至少：4KB

(2)图形方式下：

颜色 216 种需用 16 位 (2B) 表示

显示屏上共有： $1024 \times 1024 = 220$ 个点

每个点所属性需 2B 表示

则所需的属性显存： $220 \times 2B = 221B = 2MB$

显存容量至少：2M

注意：在图形方式下无基本显存

三、计算题的例题

1、原码的一位乘法

见上面复习重点的第四章 CPU 的第 4 点

2、循环校验码

见上面复习重点的第三章存储器的第 4 点

3、将内存地址转入高速缓存

现有一个 cache，其容量为 1kB，其页大小为 128B，若 cache 与内存的地址映像方式为直接映像，且内存编址方式为字节，现有一内存单元地址为 1a2b3cH，若该单元数据已调入 cache，试问：该单元在 cache 中的地址(页号和页内地址)。