

计算机组成原理

一. 名词解释 (10*2' =20')

1. **硬件**：是指可以看得见、摸得着的物理设备（部件）实体，一般讲硬件还应包括将各种硬件设备有机组织起来的体系结构。

软件：程序（代码）+ 数据 + 文档。由两部分组成，一是使计算机硬件能完成运算和控制功能的有关计算机指令和数据定义的组合，即机器可执行的程序及有关数据；二是机器不可执行的，与软件开发、过程管理、运行、维护、使用 and 培训等有关的文档资料。

固件：将软件写入只读存储器 ROM 中，称为固化。只读存储器及其写入的软件称为固件。固件是介于硬件和软件之间的一种形态，从物理形态上看是硬件，而从运行机制上看是软件。

2. **机器字长**：又称处理机字长，指处理机运算中一次能够完成二进制运算的位数，如 32 位机、64 位机。

3. **指令流**：在取指周期中从内存中读出的信息流称为指令流，它通过总线、CPU 内部数据通路流向控制器。

数据流：在执行周期中从内存中读出的信息流称为数据流，它通过总线、CPU 内部数据通路流向运算器。

4. **cache**：高速缓冲存储器，高速小容量半导体存储器，强调快速存取指令和数据，比主存速度快但存储容量小，介于 CPU 和主存之间。

5. **虚拟存储器**：是通过硬件/操作系统，实现主存-外存之间的信息部分调入调出，为用户提供了一个比实际物理内存容量大得多的存储器逻辑空间，使之为更大或更多的程序所使用。主存-外存之间的信息部分调入调出过程对用户透明。

6. **指令系统**：一台计算机中所有机器指令的集合，称为这台计算机的指令系统。指令系统是表征一台计算机性能的重要因素，其格式与功能直接影响机器的硬件

结构、软件、适用范围等。

7.寻址方式：告诉计算机如何获取指令和运算所需要的操作数。即如何提供将要执行的指令所在存储单元的物理地址；如何提供运算所需要的操作数所在存储单元的物理地址、或者操作数所在内部寄存器的编号。

8.CISC：指令条数多、结构形式复杂多样、寻址方式种类繁多、功能复杂多样、翻译执行效率低、很多指令难得用到。CISC 使计算机的研制周期长，难以保证正确性，不易调试、维护，大量使用频率很低的复杂指令浪费了系统硬件资源。

RISC：选取使用频率最高的一些简单指令，指令条数少，复杂功能通过宏指令实现；指令长度、格式、结构形式、寻址方式种类少，翻译执行效率高；只有取数/存数指令访问存储器，其余指令的操作均在 CPU 内部寄存器之间进行。RISC 可缩短计算机的研制周期、易于保证正确性、调试、维护，系统硬件资源使用效率高。

9.微命令：控制器通过控制线向部件发送的各种控制信号/操作命令。

微操作：部件接收微命令以后所完成的操作，微操作是执行部件中最基本的、不可再分解的操作。

10.总线定时：一次总线操作由若干的事件（基本操作）组成，而且这些事件间具有一定的时序关系，即一个事件什么时候开始、多长时间内完成、事件的先后顺序关系。

11.总线仲裁：当总线上的多个主设备（主方）同时竞争使用总线时，必须通过总线仲裁部件，以某种方式和策略选择其中一个主设备（主方），接管总线的控制权，传送信息。

12.DMA：直接内存访问方式，是一种完全由硬件（称为 DMA 控制器）执行、在 I/O 设备和内存之间直接交换批量信息的工作方式，包括 DMA 请求、DMA 响

应、DMA 传输、DMA 结束处理等步骤。在 DMA 传输过程中，DMA 控制器从 CPU 接管总线控制权，向内存发出地址和控制信号、修改地址、计数、以中断方式向 CPU 报告 DMA 传输结束，数据交换不经过 CPU，而直接在内存和 I/O 设备之间进行。

13.通道：是一个特殊功能的处理器（基于微处理器 CPU、单片机实现的），它有自己的指令和程序（通道指令、通道程序）专门负责数据输入/输出的传输控制，而 CPU 将“传输控制”的功能下放给通道后只负责“数据处理”功能。CPU 和通道分时使用系统总线和存储器，实现了 CPU 内部运算与 I/O 设备的并行工作。

14.指令流水线：指指令执行步骤的并行。将指令流的处理过程划分为取指令、指令译码、执行、写结果等几个并行处理的过程段。

算术流水线：是指运算操作步骤的并行，例如流水乘法器、流水除法器、流水浮点运算器等。

处理机流水线：又称为宏流水线，是指程序步骤的并行。由一串级联的处理机构成流水线的各个过程段，每台处理机负责某一特定的任务。

16.SMP：对称多处理机 SMP，结构与功能相同的多处理机通过总线或专用网络连接在一起，共享主存和 I/O 设施，由集中操作系统统一管理，提供各处理机的作业/任务调度、以及处理机间的同步，使得多个处理机的存在对用户是透明的，感觉就是一个处理机。

17.机群系统：是由一组完整的计算机（指离开机群系统仍能独立运行自己任务，有自己的操作系统、内存、I/O 设施，一般称为节点），通过标准通信协议、商品化网络（易于连接应用）连接在一起，在机群软件（中间件）统一管理调度下，可作为一个统一的计算资源协同工作，使用户感觉就是一个计算机系统。

二 . 填空 (10*2' =20')

1.CPU 执行时间 Tcpu、CPI、MIPS、FLOPS :

CPU 执行时间 : 表示 CPU 执行一段程序所占用的 CPU 时间

$$\text{CPU 时间} = \text{CPU 时钟周期数} \times \text{CPU 时钟周期长}$$

CPI : 执行一条指令所需要的平均时钟周期数

$$\text{CPI} = \frac{\text{执行某段程序所需的 CPU 时钟周期数}}{\text{该程序包含的指令条数}}$$

MIPS : 平均每秒执行多少百万条定点指令数

$$\text{MIPS} = \text{指令条数} / (\text{程序执行时间} \times 10^6)$$

FLOPS : 平均每秒执行浮点操作的次数 , 用来衡量机器浮点操作的性能

$$\text{FLOPS} = \text{程序中的浮点操作次数} / \text{程序执行时间 (秒)}$$

指令类型	指令数	时钟周期数
定点运算	25000	1
逻辑运算	20000	1
数据传送	20000	2
程序控制	12000	2
浮点运算	10000	2
字符串处理	5000	2
输入输出	8000	2

[例题]用一台 40MHz 处理机执行标准测试程序 , 它包含的混合指令数和相应所需的时钟周期数如下表所示。求 : CPI、MIPS、Tcpu。

$$\text{CPI} = ((25000 + 20000) * 1 + (20000 + 12000 + 10000 + 5000 + 8000) * 2) / (25000 + 20000 + 20000 + 12000 + 10000 + 5000 + 8000)$$

$$= 1.55 \text{ (周期/指令)}$$

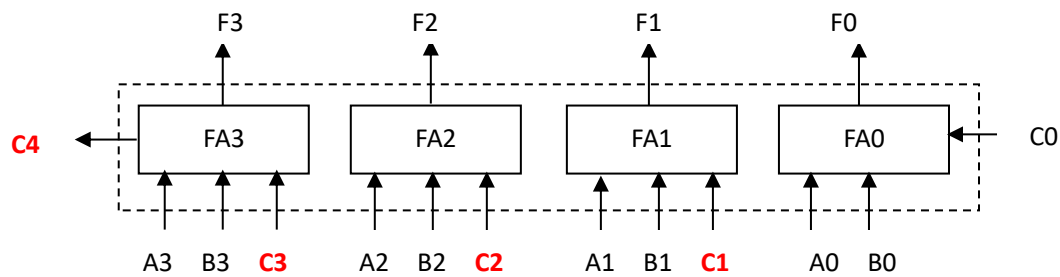
$$\text{MIPS} = f / (\text{CPI} \times 10^6)$$

$$= 40 \times 10^6 / (1.55 \times 10^6) = 25.81 \text{ (百万条指令/秒)}$$

$$\text{Tcpu} = ((25000 + 20000) * 1 + (20000 + 12000 + 10000 + 5000 + 8000) * 2) / (40 * 10^6)$$

$$= 3.875 \times 0.001 \text{ (秒)}$$

2.并行进位加法器中的进位：



(1) 设被加数 $A=A_3A_2A_1A_0$ ，加数 $B=B_3B_2B_1B_0$ ，且有：

$$G_0=A_0B_0 \quad P_0=A_0+B_0$$

$$G_1=A_1B_1 \quad P_1=A_1+B_1$$

$$G_2=A_2B_2 \quad P_2=A_2+B_2$$

$$G_3=A_3B_3 \quad P_3=A_3+B_3$$

G_i ：进位产生函数，表示两个数位都为 1

P_i ：进位传递函数，表示某位上的两个数位有一个为 1，如果来自低位的进位为 1，则肯定会产生进位。

(2) 串行加法器的串行进位方式：

$$C_1=G_0+P_0C_0$$

$$C_2=G_1+P_1C_1$$

$$C_3=G_2+P_2C_2$$

$$C_4=G_3+P_3C_3$$

(3) 并行加法器的并行进位方式：所有进位可以同时产生，实际上只依赖于数位本身、来自最低位的进位 C_0 。

$$C_1=G_0+P_0C_0$$

$$C_2=G_1+P_1C_1=G_1+P_1(G_0+P_0C_0)=G_1+P_1G_0+P_1P_0C_0$$

$$C_3=G_2+P_2G_1+P_2P_1G_0+P_2P_1P_0C_0$$

$$C_4=G_3+P_3G_2+P_3P_2G_1+P_3P_2P_1G_0+P_3P_2P_1P_0C_0$$

3.流水线加速比：

[例题]在 4 级流水线加法器中 ,(1)假设每个过程段所需的时间为：求阶差 $\tau_1 = 70\text{ns}$,对阶 $\tau_2 = 60\text{ns}$,相加 $\tau_3 = 90\text{ns}$,规格化 $\tau_4 = 80\text{ns}$,缓冲寄存器 L 的延时为 $t_l = 10\text{ns}$,求 4 级流水线加法器的加速比为多少？(2)如果每个过程段的时间相同,即都为 75ns (包括缓冲寄存器时间),加速比是多少？

(1) 加法器的流水线时钟周期为： $\tau = 90\text{ns} + 10\text{ns} = 100\text{ns}$

不采用流水线方式,则浮点加法器完成一个任务所需的时间为： $\tau_1 + \tau_2 + \tau_3 + \tau_4 = 300\text{ns}$

因此,4 级流水线加法器的加速比为： $C_k = 300/100 = 3$

(2) 当每个过程段的时间都是 75ns 时,加速比为

$$C_k = (75 \times 4) / 75 = 4$$

注：与加速比定义有点不一致，理解：当任务数 n 很大时，采用流水线一个时钟周期完成一个任务，而非流水线 $\tau_1 + \tau_2 + \tau_3 + \tau_4$ 完成一个任务

4.存储器带宽：

单位时间内从存储器读出的二进制数信息量，一般用 B/s （字节/秒）表示。

[例题]设存储容量为 64M ，字长为 64 位，模块数 $m=8$ ，分别用顺序和交叉方式进行组织。存储周期 $T=100\text{ns}$ ，数据总线宽度为 64 位，总线传送周期 $t=50\text{ns}$ 。求：顺序存储器和交叉存储器的带宽各是多少？

[解]以读出 8 个字进行考虑。

信息总量： $q=64 \times 8=512\text{bits}$

顺序存储器读出 8 个字的时间： $t_1=mT=8 \times 100\text{ns}=8 \times 10^{-7}\text{s}$

交叉存储器流水读 8 个字时间： $t_2=T+(m-1)t$

$$=100\text{ns}+(8-1) \times 50\text{ns}$$

$$=450\text{ns}=4.5 \times 10^{-7}\text{s}$$

顺序存储器的带宽： $W_1=q/t_1=512/(8 \times 10^{-7}\text{s})=64 \times 10^7\text{bps}$

交叉存储器的带宽： $W_2=q/t_2=512/(4.5 \times 10^{-7}\text{s})=114 \times 10^7\text{bps}$

5.cache/主存系统的平均访问时间：

[例题] CPU 执行一段程序时，cache 完成存取的总次数为 2420 次，主存完成存取的总次数为 80 次，已知 cache 存储周期为 40ns，主存存储周期为 240ns，求 cache/主存系统的效率和平均访问时间。

[解]

$$h = N_c / (N_c + N_m) = 2420 / (2420 + 80) = 2420 / 2500 = 0.968$$

$$r = t_m / t_c = 240 / 40 = 6$$

$$e = 1 / (r + (1 - r)h) = 1 / (6 + (1 - 6)0.968) = 84.7\%$$

$$t_a = t_c / e = 40 / 0.847 = 47\text{ns}$$

6.三级时序体制：

指令周期：取出一条指令并执行这条指令所需要的时间。

CPU 周期：常称为机器周期，每条指令的执行过程可划分为若干个阶段，如取指令、取源操作数、取目的操作数、执行运算、保存结果等阶段，每个阶段所对应的时间。

时钟周期：每个阶段由若干有序的、不可再分的基本操作过程构成，如从数据 cache 中取一个源操作数送到 R0，包括地址→AR、AR→M、M→DR、DR→R0 等基本操作过程，每个操作过程所对应的时间。

显然，**指令周期**对应一条**指令**，**CPU 周期**对应一个**阶段**，**时钟周期**对应一个**基本操作过程**。因此，时间应按**指令周期**、**CPU 周期**、**时钟周期**进行分级控制，这就是三级时序体制。

7.三级存储系统：

目前在计算机系统中通常采用多级存储器体系结构，即**高速缓冲存储器、主存储器和外存储器**。

CPU 能直接访问高速缓冲存储器 cache 和内存；外存信息必须调入内存后才能为 CPU 进行处理。

(1)高速缓冲存储器：高速小容量半导体存储器，强调快速存取指令和数据；

(2)主存储器：介于 cache 与外存储器之间，用来存放计算机运行期间的大量程序和数据。要求选取适当的存储容量和存取周期，使它能容纳系统的核心软件和较多的用户程序；

(3)外存储器：大容量辅助存储器，强调大的存储容量，以满足计算机的大容量存储要求，用来存放系统程序、应用程序、数据文件、数据库等。

8.CPU 的功能：

(1)指令控制

控制程序的执行顺序；

由于程序是一个指令序列，这些指令的相互顺序不能任意颠倒，必须严格按照程序规定的顺序进行。

(2)操作控制

控制器产生取指令、执行指令的所需要的全部操作控制信号，并依序送往相应的部件，从而控制这些部件按指令的要求完成规定的动作。

(3)时间控制

对各种操作实施时间上的定时；

在计算机中，各种指令的操作信号和整个执行过程均受到时间的严格定时和事件先后顺序控制（应在规定的时间点开始，在规定的时间内结束），以保证计算机有条不紊地自动工作。

(4)数据加工

完成指令规定的运算操作。

9.流水线的相关问题：

(1)资源相关：是指多条指令进入流水线后，在同一流水线时钟周期内争用同一个功能部件所发生的冲突。

(2)数据相关：在一个程序中，如果必须等前一条指令执行完毕以后，才能执行后一条指令（即后一条指令需要引用前一条指令的结果数据），那么这两条指令就是数据相关的。

(3)控制相关：控制相关冲突是由转移类指令引起的。当执行转移类指令时，可能为顺序取下条指令；也可能转移到新的目标地址取指令。如果流水线顺序取指令，而程序却需要转移时，进入流水线的指令并不是将要执行的指令，或者转移的目标指令可能还没有进入流水线，从而使流水线发生断流。

10.体系结构的分类：

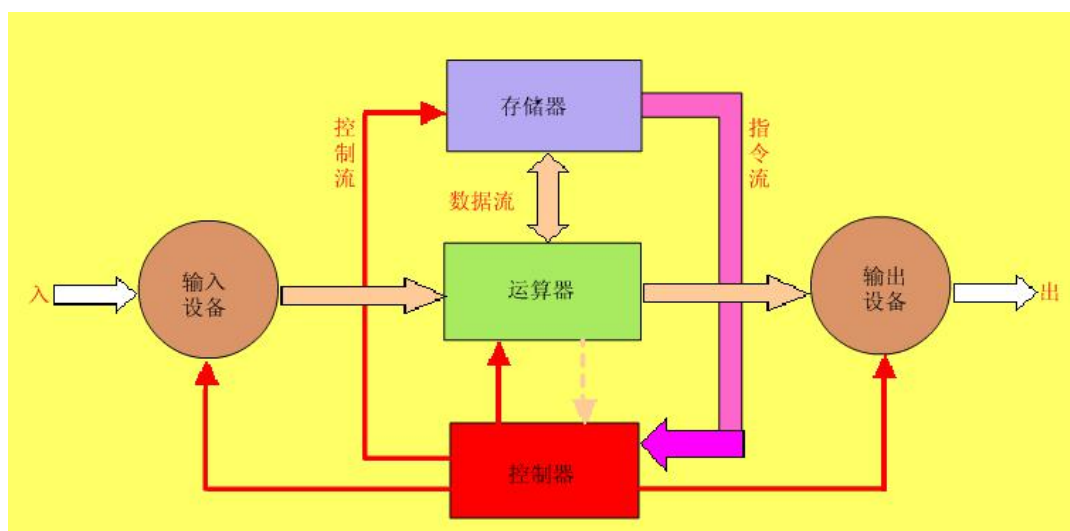
（1）单指令流单数据流 SISD：传统单处理机系统

（2）单指令流多数据流 SIMD：向量处理机

（3）多指令流单数据流 MISD：没意义

（4）多指令流多数据流 MIMD：多处理机系统、机群系统

11.冯诺依曼型计算机的主要设计思想：



(1)由运算器、控制器、存储器、输入设备、输出设备五大部分构成计算机硬件系统概念结构；

(2)采用二进制代码表示数据和指令； (3)采用存储程序控制方式（指令驱动）。

三 . 专题 ($6 \times 10' = 60'$)

1.浮点加减法 (对阶、尾数加、规格化、溢出、舍入)

2.存储器设计

3.寻址方式 指令格式

4.指令周期流程图

5.多总线结构

6.中断 DMA

7.cache 基本原理

8.微程序控制器原理