

第1章 计算机系统结构导论

• 1.1 计算机系统的基本概念

• 1.1.1 计算机系统的组成

- 由运算器、控制器、存储器、输入输出设备组成
- 硬件和软件
软件：应用，可编程硬件上的软件...
- 由 人员、数据、设备、程序和规程 组成

• 1.2 计算机系统的发展

• 1.2.1 Von Neumann 结构的特点

存储指令，按地址访问；指令的编码方式；数据和指令被同等对待

• 特点

- 1) 存储器是按地址访问的顺序线性编址的一维结构，每个单元的位数是固定的
- 2) 指令由操作码和地址码组成
- 3) 指令在存储器中是按其执行顺序存储的，有程序计数器指明每条指令所在单元的地址
- 4) 在存储器中指令和数据被同等对待
- 5) 计算机系统几个以运算器、控制器为中心

瓶颈 在存储器

- 6) 指令、数据均已二进制编码表示，采用二进制运算
- ** 哈弗结构：数据和指令 是分开存储的

• 局限性

- 1) 以数值计算为主，对图像、图形、音频等处理较差
- 2) 程序顺序型，限制并行操作，运算速度无法突破
- 3) 软件系统越来越复杂，正确性无法保证，软件生产率低下
- 4) 硬件投资较大，可靠性差，发展受限
- 5) 编程人员专业性高

• 1.2.2 器件发展对系统结构的影响

- 1. 器件的功能和使用方法对系统结构的影响
- 2. 器件的发展对系统结构的影响

• 1.2.3 应用对系统结构的影响

• 1.2.4 算法对系统结构的影响

• 1.2.5 价格对系统结构的影响

- 高性价比

• 1.2.6 功耗对系统结构的影响（略）

• 1.3 计算机系统的功能和结构（重点）

• 1.3.1 层次结构

- 1. 功能模型

- 计算机系统 = 硬件 + 软件；存储过程 -> 指令过程
- 一个信息处理过程，可用控制流程来描述控制流程的实现。
- 实现的方式：
 - 1) 全硬件：组合逻辑
 - 2) 软 + 硬：微程序 + 硬逻辑
 - 3) 全软件：程序
- 功能模型图

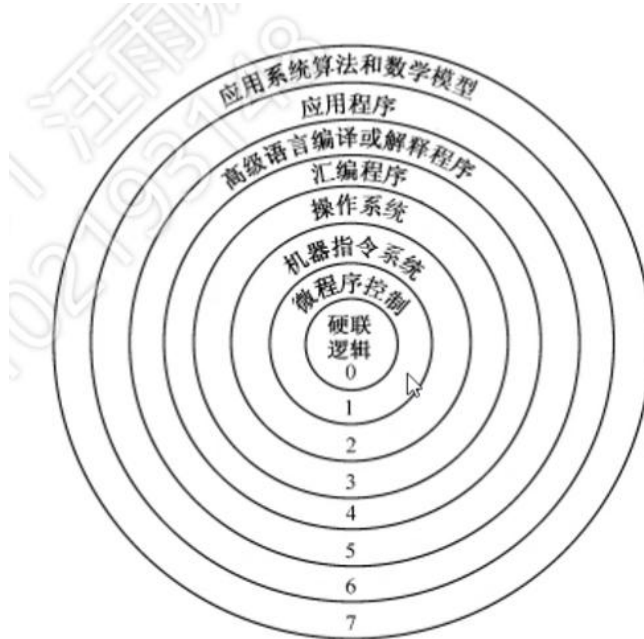
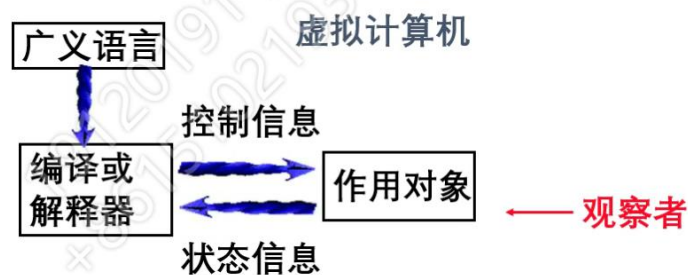


图1-5 计算机系统功能模型

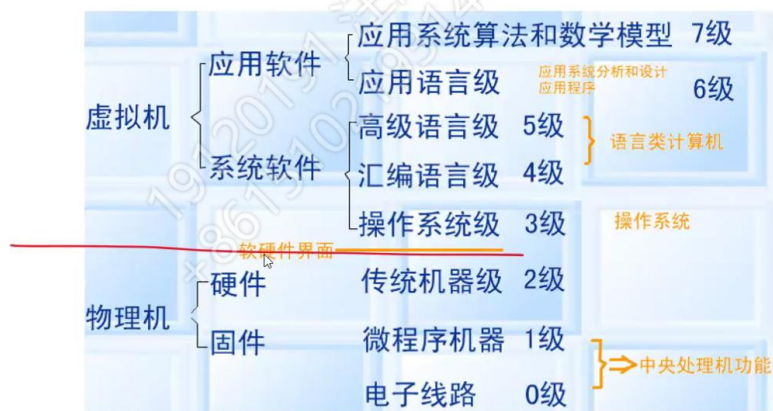
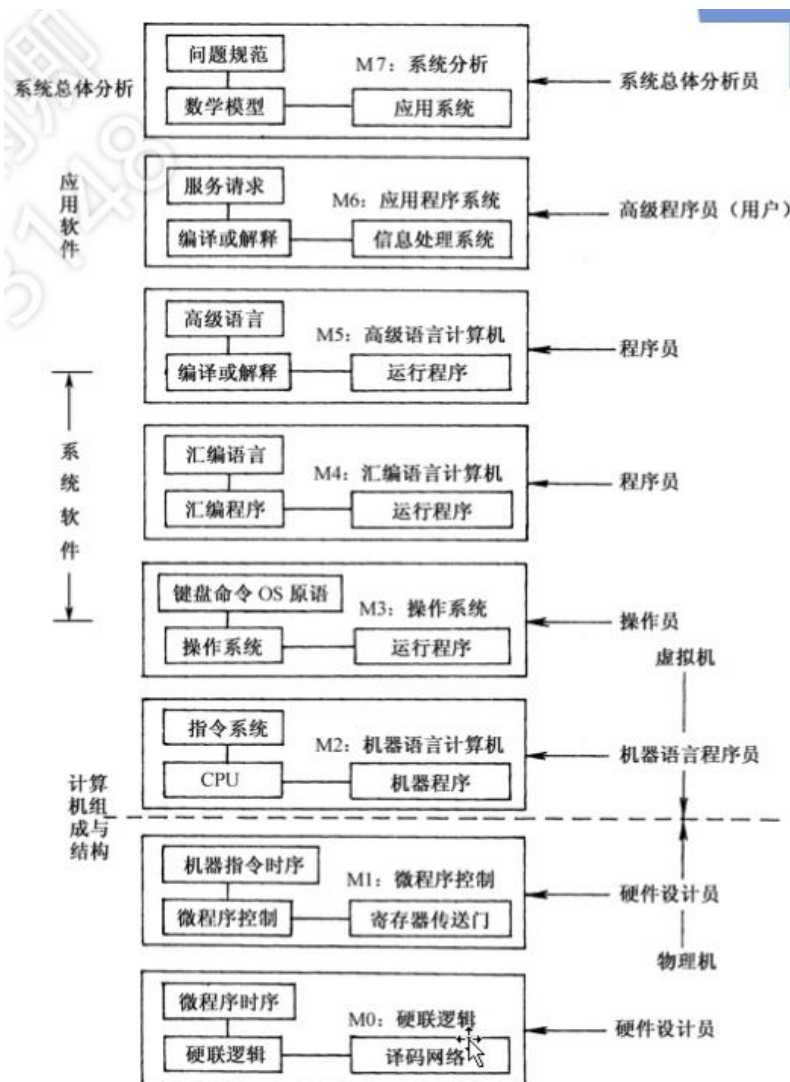
• 2. 层次结构

- 计算机使用语言：
 - 微指令=》机器指令=》OS 指令=》汇编语言=》高级语言=》应用语言
- 虚拟机：有软件实现的机器（此处不包含硬件使用的语言）



观察者通过广义语言和对象交互；虚拟机是功能模型的抽象

- 层次图（8 个层级）



• 1.3.2 定义

- 计算机系统结构是程序员所看到的计算机属性，即概念性结构与功能属性
- 计算机系统中对各级之间界面的定义及其上、下的功能分配（普遍认可）
- 透明性
- 重点：指令系统

• 1.3.3 计算机组成与实现

- 系统结构 vs 组成

1.3.3 计算机组成与实现

*.计算机系统结构与组成差别:

例1: 确定指令系统功能--系统结构。

指令的实现: 取指, 取操作数, 运算, 送结果--组成。

例2: 确定有乘, 除法指令--系统结构。

用乘法器还是用加法器右移加实现--组成。

例3: 确定MEM容量, 编址方式 (按bit, byte, word访问)
--系统结构。

MEM速度, 逻辑结构, 性能价格比--组成。

系统结构: 确定有什么指令功能

- 组成 vs 实现

*.组成与实现差别:

例1: 乘法指令用乘法器还是用全加器右移加--组成。

乘法器或加法器用什么器件线路及工艺技术--实现。

● 1.3.4 计算机系统结构、组成和实现三者的关系

- 系统结构: 计算机系统的软硬件界面区分

哪些是软件做, 哪些是硬件做

- 组成: 计算机系统结构的逻辑实现

- 计算机实现: 计算机组成的物理实现

● 1.3.5 计算机系统的特性

- 1. 计算机等级 (了解)

- 2. 系列机概念

- 定义: 利用事先设计好的系统结构, 按器件状况和硬件技术研究实现方法, 并按速度、价格等不同要求配置各档机器。

- 数据表示方式一致, 系统软件兼容

例: 计组浮点中表示, 都用 32 位表示。(对于高中低所有机型)

- 软件兼容 4 个方向: 上、下、前、后

软件兼容: 同一个软件可以不加修改地运行于系统结构相同的各机器上, 而所得结果一致。

- 向下兼容: 高档机不加修改运行于低档机 (很少)

- 向上兼容: 低档机不加修改运行于高档机。(基本实现)

- 向后兼容: 几年前的软件可以在现在的机器上运行。(基本实现)

- 3. 模拟与仿真

- 模拟

- 模拟 (Simulation)

B虚拟机 (Virtual Machine) A宿主机 (Host Machine)

B的一条机器指令用A的一段机器语言程序去解释执行-->模拟。

A B 机器在同一层级实现

- **仿真**

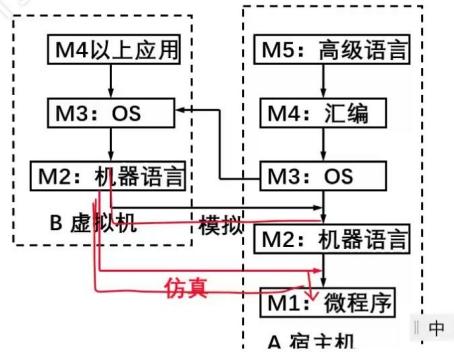
B目的机 (Target Machine) A宿主机 (Host Machine)
B的一条机器指令用A的一段微程序去解释执行--> **仿真**。

A 即本来的一条指令，也是用一段微程序实现。

- 模拟、仿真、系列机

模拟和仿真目的：

为了实现不同
系统结构的机
器间的软件移
植



模拟在同一层级实现；仿真是在不同层级实现；系列机可以实现软件兼容。（在指令格式不同的情况下，只能通过仿真和模拟实现软件兼容）

• 1.4 计算机系统设计的方法

- 目标：高性价比



- 1.4.2 计算机系统设计的定量原则

- 1. 加快经常性事件的速度 (核心)

- 2. Amdahl's Law

- 定义

- 2、Amdahl's Law

Amdahl's Law states that the performance improvement to be gained from using some faster mode of execution is limited by the fraction of the time the faster mode can be used.
系统中某部件因采用某种更快执行方法后整个系统性能的提高与这种执行方式使用频率或占总执行时间的比例有关

加速比 = 采用改进措施后的性能 / 采用改进措施前的性能 = 采用改进措施前执行某任务的时间 / 采用改进措施后执行某任务的时间

加速比 = $\frac{\text{采用改进措施后的性能}}{\text{采用改进措施前的性能}} = \frac{\text{采用改进措施前执行某任务的时间}}{\text{采用改进措施后执行某任务的时间}}$
 加速比与两个因素有关：

比率 $F_e = \frac{\text{可改进部分占用时间}}{\text{改进前整个任务执行时间}} \quad (F_e < 1)$
 $S_e = \frac{\text{改进前改进部分执行时间}}{\text{改进后改进部分执行时间}} \quad (S_e > 1)$

●结论

① 改进后整个任务的执行时间：

$$T_n = T_o \left[(1 - F_e) + \frac{F_e}{S_e} \right]$$

式中， T_o 为改进前整个任务的执行时间。

② 改进前后整个系统的加速比：

$$S_n = \frac{T_o}{T_n} = \frac{1}{(1 - F_e) + \frac{F_e}{S_e}}$$

●例题

例1-2:

设求浮点数平方根FPSQR操作占整个测试程序执行时间的20%。一种实现方法是采用FPSQR硬件，使其速度加快到10倍。另一种实现方法是使所有浮点数指令FP速度加快到2倍，同时，设FP指令占整个程序执行时间50%。请比较两种方案优劣。

解答： 硬件方案 $F_e = 0.2$ ， $S_e = 10$ ，则

$$S_n = \frac{1}{(1 - 0.2) + \frac{0.2}{10}} = \frac{1}{0.82} = 1.22$$

FP 加速方案 $F_e = 0.5$ ， $S_e = 2$ ，则

$$S_n = \frac{1}{(1 - 0.5) + \frac{0.5}{2}} = \frac{1}{0.75} = 1.33$$

● 1.4.4 计算机系统的设计步骤

- 中间（软/硬件界面）开始，合理分配软硬件功能

● 公式例题

四. 计算题：

1. 现需要在双核处理器上对两个程序进行优化工作。假定对该程序进行并行化时，该部分的加速比为2。请问：

- (1) 假定程序A的40%可以并行化，那么在隔离运行时，通过这个应用程序可以实现多大的加速比？
- (2) 假定程序B的99%可以并行化，那么在隔离运行时，这个应用程序可以实现多大的加速比？

←

2. 当一个计算运行于向量硬件的向量模式时，其速度是正常执行模式的10倍。

我们将使用向量模式时花费的时间百分比称为向量化百分比。那么当向量化百分比达到多少时，才能使加速比为2？

- 公式： $S_n = 1 / (1 - F_e + F_e / S_e)$

- 1.1) $S_n = 1 / ((1 - 0.4) + 0.4 / 2) = 1.25$

- 1.2) $S_n = 1 / \overline{((1-0.99) + 0.99 / 2)} = 1.98$
- 2.1) $S_e = 10, F_e = ? , S_n = 2$
 - $2 = 1 / (1 - F_e + F_e / 10)$
 - $F_e = 56\%$

• 1.5 现代计算机系统结构的研究领域

• 1.5.1 计算机系统结构分类

• Flynn 分类法

- 1) 指令流：机器执行指令序列
- 2) 数据流：指令调用的数据序列
- 3) 多倍性：在系统最受限制的部件上，同时处于同一阶段的指令或数据的最大个数

• 具体分类

• SISD: 单指令单数据 (早期使用)

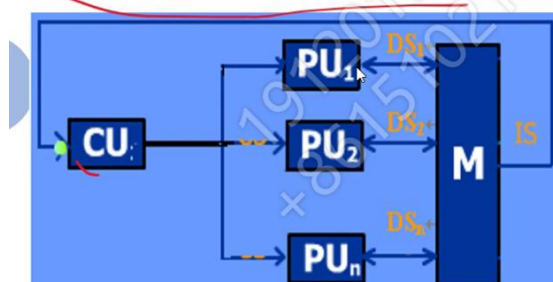
SISD: 单指令流单数据流



串行计算机：有一个控制部件管理

• SIMD (经常使用)

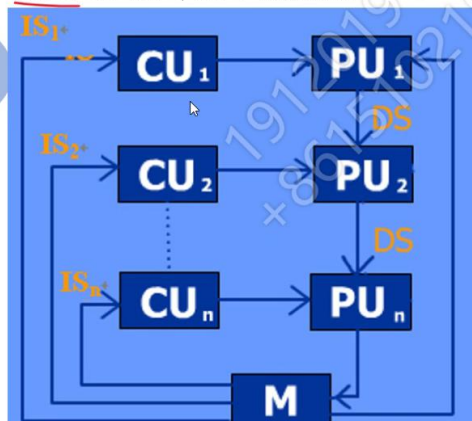
SIMD: 单指令流多数据流



并行处理机（阵列机）：由一个控制部件管理，多个 PU 接受 CU 来的同一条指令；
GPU；CPU 扩展

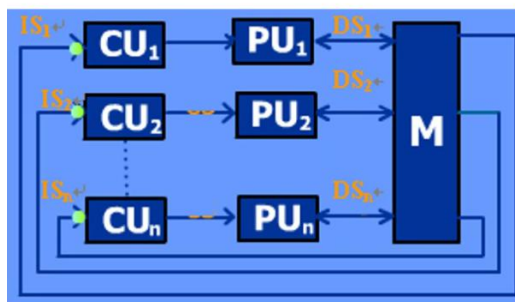
• MISD (不常使用)

MISD:多指令流单数据流



- **MIMD**

MIMD:多指令流多数据流

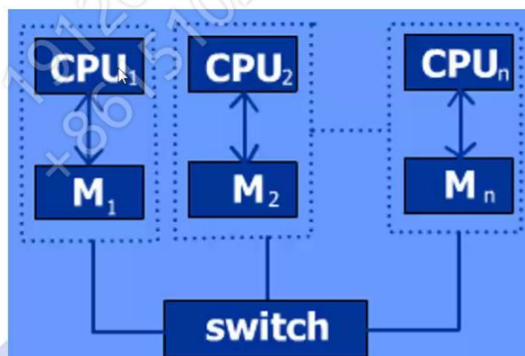


多处理机和多计算机

- **SPMD (了解)**

SPMD:单程序多数据流

*.基于Cluster



CPU+M1 作为一个整体;

- **SIMT: 单线程多数据流 (了解)**

- **1.5.2 现代计算机系统结构研究方向**

- **1.5.3 计算机系统结构发展趋势**

- **第一章 习题**

- 填空

一. 填空题:

1. 冯·诺依曼型计算机系统结构是由____、____、____和____五个部分组成。
2. 计算机系统由____和____两部分组成，二者是不可分割的整体。
3. 信息的处理流程可以用控制流程的概念来描述，分为____、____、____。
4. 符号化的机器指令被称为汇编语言，在上层则是用户通用的____。
5. ____是计算机系统的软、硬件界面，____是计算机系统结构的逻辑实现，____是计算机组成的物理实现。
6. 为了满足时序的要求，对于使用频率高的模块，使用供电电压较____（高/低）的电源，以降低时序路径中单元的延迟，从而降低整条时序路径的延迟。
7. Amdahl 定律认为：系统中某部件采用某种更快的执行方法后，整个系统性能的提高与这种执行方法的____或____有关。
8. SISD 系统结构主要表示了大多数____计算机，SIMD 系统结构以____为代表。

- 1. 控制器、运算器、存储器、输入输出设备
 - 2. 软件；硬件
 - 3. 全硬件；硬件和软件；全软件
 - 4. 高级语言
 - 5. 计算机系统结构；计算机组成；计算机实现
 - 7. 使用频率；占总执行时间的比例
 - 8. 串行；并行计算机，GPU
- 选择

二. 选择题:

1. 按照冯·诺依曼计算机体系结构来说，计算机的运算速度与（ ）有关。
 - A. 访问存储器的次数
 - B. 输入/输出设备性能
 - C. 数据编码方式
 - D. 指令的组成方式
2. 从计算机系统结构上讲，机器语言程序员所看到的机器属性是（ ）。
 - A. 计算机各部件的硬件实现
 - B. 计算机软件所要完成的功能
 - C. 编程要用到的硬件组织
 - D. 计算机硬件的全部组成
3. 机器指令又称机器语言，是面向（ ）的最基层语言。
 - A. 系统
 - B. 硬件
 - C. 软件
 - D. 用户
4. 按照功能和使用方法，以下不属于芯片种类划分的是（ ）。

- A. 用户片
- B. 非用户片
- C. 现场片
- D. 非现场片

6. 信息的处理过程可用控制流程的概念来描述。以下不属于控制流程的实现方法的是（ ）。[❏]
- A. 全硬件法 B. 硬件与软件相结合的方法[❏]
C. 全软件法 D. 半硬件与半软件法[❏]
7. Amdahl 提出：计算机系统结构是从（ ）所看到的计算机的属性，即概念性结构和功能特性，这实际上是计算机系统的外特性。[❏]
- A. 程序设计者 B. 程序使用者[❏]
C. 硬件使用者 D. 软件使用者[❏] I
8. 计算机组成指计算机系统结构的（ ），计算机实现指计算机组成的（ ）。[❏]
- A. 物理实现；逻辑实现 B. 逻辑实现；物理实现[❏]
C. 都是物理实现 D. 都是逻辑实现[❏]
9. 一般来说，提高硬件功能比例可以（ ）运算速度，提高软件功能比例可以（ ）硬件成本。[❏]
- A. 提高；降低 B. 降低；提高[❏]
C. 都是提高 D. 都是降低[❏]

- 1. A （冯·诺依曼计算机）
- 2. C
- 3. D
- 4. D
- 6. D
- 7. A
- 8. B
- 9. A