

计算机系统结构

汪东升

wds@tsinghua.edu.cn



- ✓ ISA
- ✓ 编译技术

- ✓ 减少指令延迟
 - 层次存储

$$T = IC \times CPI \times t$$

✓ 并行化

➢ 指令级并行

□ 软件方法

- 循环展开
- 软件流水

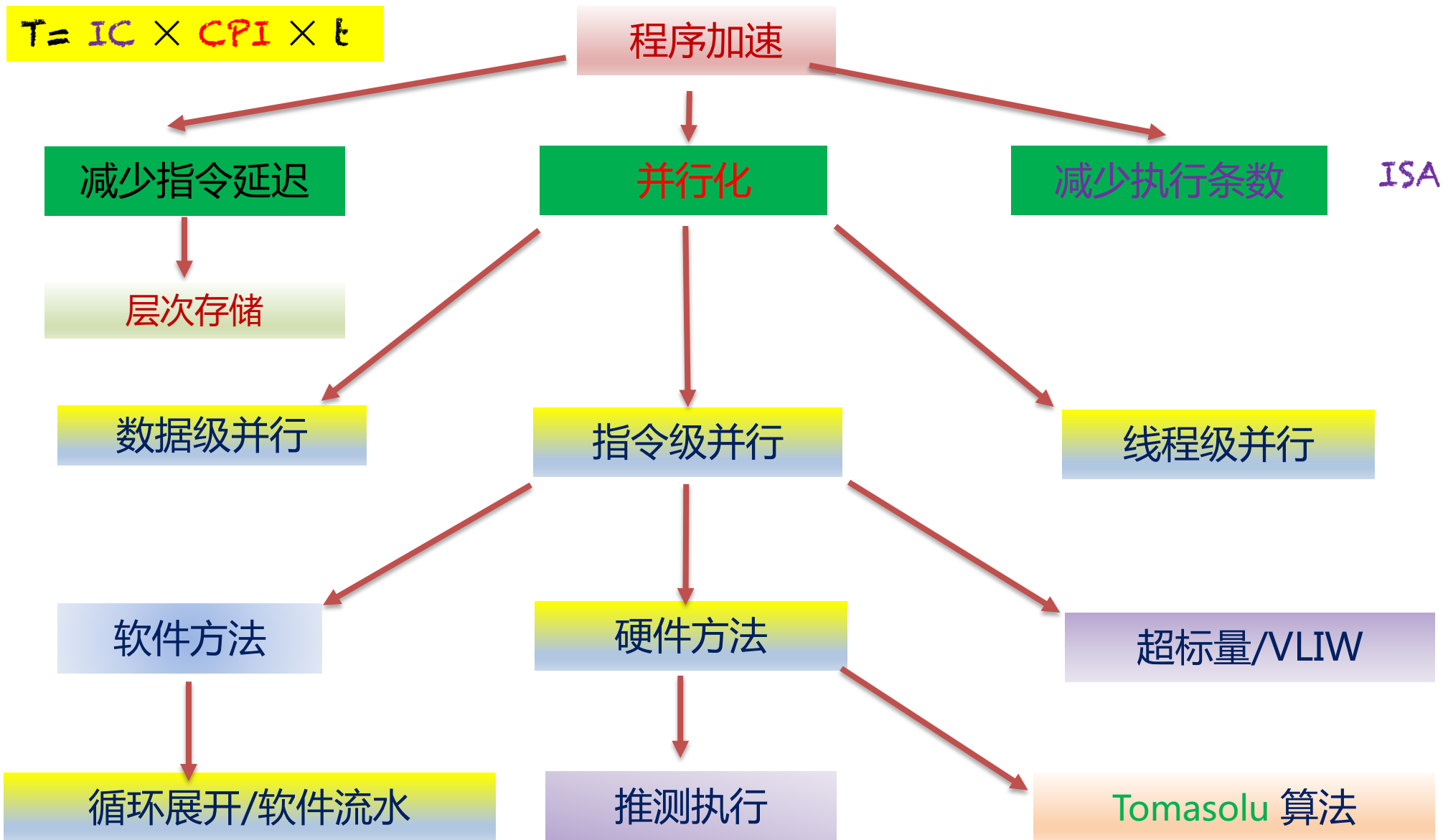
□ 硬件方法

- OOO-Tomasulo
- OOO-记分牌
- 推测执行

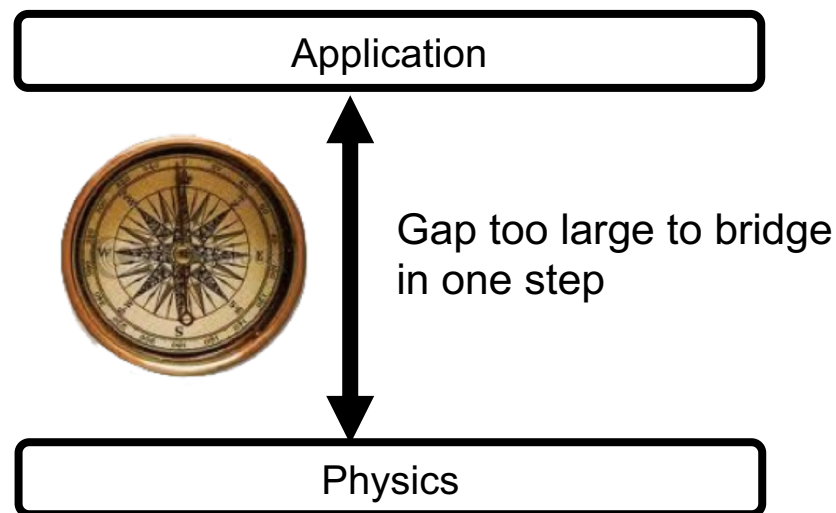
□ 超标量/VIW

- 数据级并行 (向量 GPU TPU)
- 线程级并行

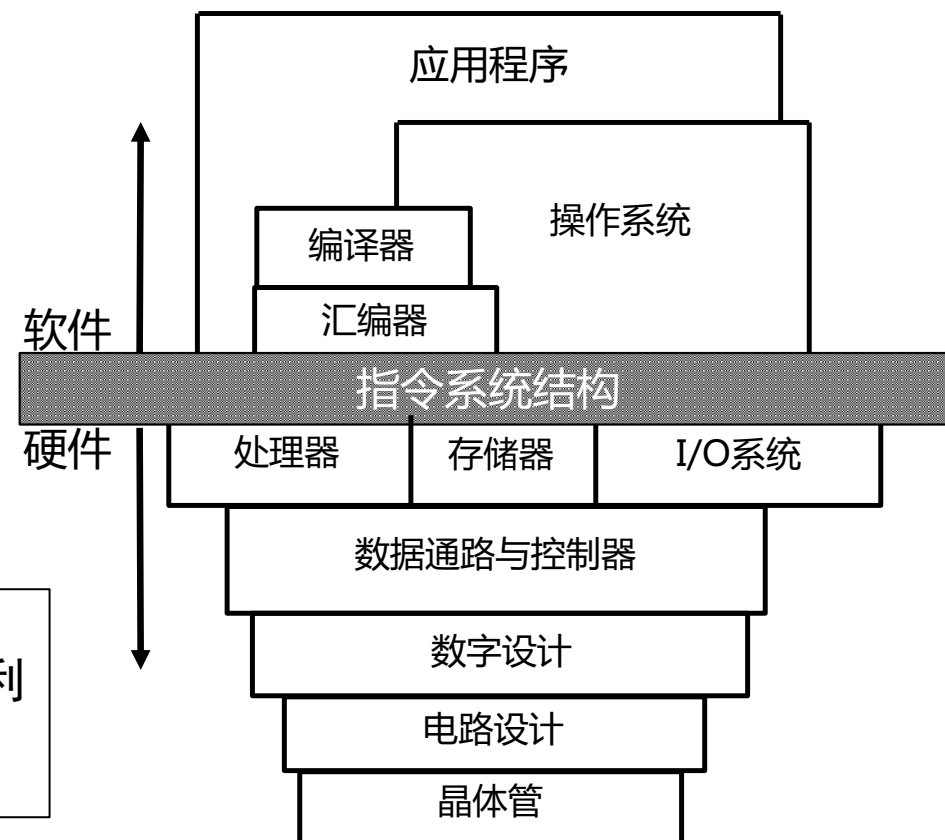
概念与性能评价



- ◆ **系统**课程：系统 = 软件 + 硬件
- ◆ 与 **计算机组成原理** 的区别 **ISA** 微架构
- ◆ 计算机系统结构的基本**概念**、基本结构、基本方法
- ◆ **优化**的设计？ **多目标**优化的负责系统 在**不确定**中寻找满足需求的优化设计
- ◆ 强调从开发**并行性**的角度，用量化分析(性能评价)的方法来分析 and 设计计算机**系统**
(性能/面积/功耗/安全性/可靠性……)
- ◆ 理论联系实际，学以致用



计算机系统结构是计算机系统抽象层的设计，使我们能够利用现有的制造技术有效地实现信息处理应用。



计算机技术的发展永远满足不了不断增长应用对计算的需求

1. 基本概念&性能评价

2. 指令系统设计

3. 流水线技术基础

4. 指令级并行（硬件方法）

5. 指令级并行（软件方法）

6. 线程级并行

7. 数据级并行（向量处理）

8. 存储系统

9. 互连网络

1. 计算机系统结构的基本概念——**层次结构**，翻译与解释，**组成与实现**，系列机与**Flynn分类法**、冯氏分类法和Handler分类法等
2. 计算机系统的设计——经常性事件原则，**Amdahl定律**，**CPU性能公式**，程序局部性原理以及由项向下等设计流程
3. 计算机系统的性能评测——**执行时间和吞吐率**，**基准测试程序**，总执行时间，算术平均时间，调和平均时间，加权执行时间及标准化执行时间等
4. 计算机系统结构的发展——冯·诺依曼结构及其改进，器件、软件与应用对系统结构的影响
5. 计算机系统结构中并行性的发展——并行性概念，提高并行性的技术途径以及并行性发展阶段总结

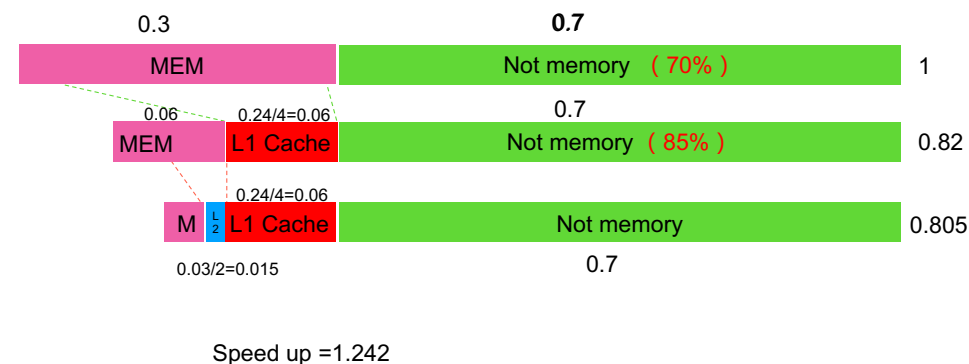
$$S_n = \frac{T_0}{T_n} = \frac{1}{(1 - F_e) + \frac{F_e}{S_e}}$$



试计算改进前后该处理器系统的加速比

假设存储器访问时间占整个程序执行时间的30%，拟对该存储系统进行性能优化：

- 第一步：采用L1 Cache，使80%存储器访问的时间缩短到原来的 $\frac{1}{4}$ ；
- 第二步：采用L2 Cache，使剩余的20%存储器访问时间缩短到原来的 $\frac{1}{2}$ 。



1. 基本概念&性能评价

2. 指令系统设计

3. 流水线技术基础

4. 指令级并行（硬件方法）

5. 指令级并行（软件方法）

6. 线程级并行

7. 数据级并行（向量处理）

8. 存储系统

9. 互连网络

1. **指令系统分类**：存储单元类型，堆栈，累加器，通用寄存器组，寄存器-寄存器结构
2. **数据类型和数据表示**：数据类型，数据表示，确定原则，操作数大小，访问频率
3. **寻址方式**：整数边界，立即数寻址，直接寻址，寄存器寻址，寄存器间接寻址，寄存器相对寻址，基址变址寻址，相对基址变址寻址
4. **数据操作**：数据传送类指令，算术运算指令，逻辑运算指令，程序控制指令
5. **指令系统的设计与优化**：完整性，规整性，正交性，高效率，兼容性，控制指令，**哈夫曼编码**，扩展操作码，等长扩展码，定长操作码
6. **指令系统的发展**：CISC、RISC
7. **典型指令系统结构**：RISC-V指令集、代码密度

1. 基本概念&性能评价

2. 指令系统设计

3. 流水线技术基础

4. 指令级并行（硬件方法）

5. 指令级并行（软件方法）

6. 线程级并行

7. 数据级并行（向量处理）

8. 存储系统

9. 互连网络

1. 流水线的基本概念：工业流水线，时空图，连接图，部件级、处理器级和系统级流水线，静态与动态流水线，线性与非线性流水线，顺序与乱序流水线，标量与向量流水线
2. 流水线的性能指标：吞吐率、加速比、效率
3. 非线性流水线的调度：单功能非线性流水线的调度，多功能非线性流水线的调度
4. 流水线的相关与冲突：数据相关、名相关、控制相关、结构冲突、数据冲突、控制冲突
5. 流水线的实现：RISC-V处理器，最佳段数的选择

1. 基本概念&性能评价

2. 指令系统设计?

3. 流水线技术基础

4. 指令级并行 (硬件方法)

5. 指令级并行 (软件方法)

6. 线程级并行

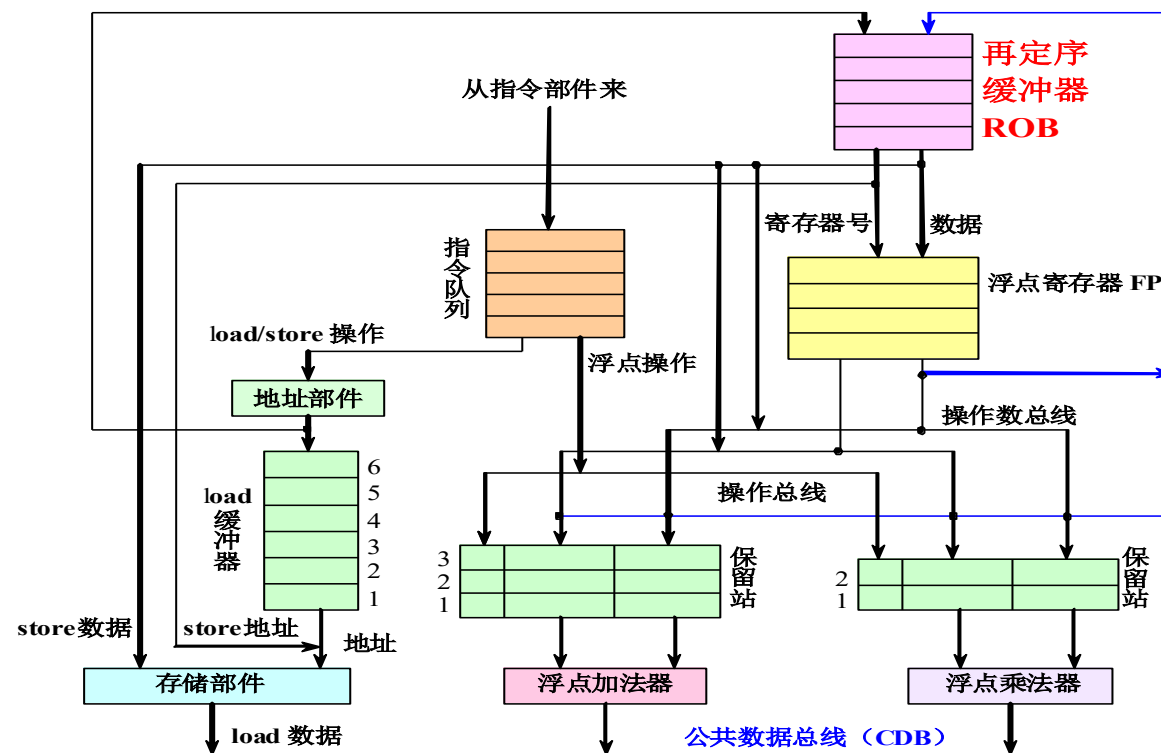
7. 数据级并行 (向量处理)

8. 存储系统

9. 互连网络

指令级并行及其开发 — 硬件方法

- 指令级并行的概念
- 掌握Tomasulo+ROB算法
- 动态分支预测技术 (1bit, 2bit)
- 推测执行



1. 基本概念&性能评价
2. 指令系统设计?
3. 流水线技术基础
4. 指令级并行 (硬件方法)
- 5. 指令级并行 (软件方法)**
6. 线程级并行
7. 数据级并行 (向量处理)
8. 存储系统
9. 互连网络

指令级并行的开发——软件方法

1. 基本指令调度及循环展开：指令调度、循环展开及注意事项
2. 跨越基本块的静态指令调度：全局指令调度、踪迹调度、踪迹选择、踪迹压缩、超块调度、尾复制技术
3. 静态多指令发射：VLIW技术：VLIW vs. 超标量、多发射处理器 vs. 向量处理器

1. 基本概念&性能评价
2. 指令系统设计?
3. 流水线技术基础
4. 指令级并行 (硬件方法)
5. 指令级并行 (软件方法)
6. 线程级并行
7. 数据级并行 (向量处理)
8. 存储系统
9. 互连网络

- 多核处理器是必然
- 对称式/分布式共享存储体系结构计算机，一致性问题是关键
- 同时多线程只有在细粒度的实现方式下才有意义
- 将超标量和同时多线程结合起来，在指令级并行基础上进一步开发线程级并行，可以获得显著的性能提高

1. 基本概念&性能评价
2. 指令系统设计?
3. 流水线技术基础
4. 指令级并行 (硬件方法)
5. 指令级并行 (软件方法)
6. 线程级并行
- 7. 数据级并行 (向量处理)**
8. 存储系统
9. 互连网络

向量处理机

- 向量的三种处理方式;
- 向量处理机的结构;
- **提高向量处理机性能的常用技术;**
 - ✓ 向量链接
 - ✓ 编队
 - ✓ 循环开采
- **向量处理机的性能评价方法。**

- 存储系统层次架构

1. 基本概念&性能

- 局部性原理 (Principle of locality)

2. 指令系统设计

- 层次存储的 4 个基本问题

- 放置策略: $y=f(x)$

3. 流水线技术基础

- 查找策略: $y=f(a)$

4. 指令级并行 (硬件方法)

- 替换策略
- 写策略

5. 指令级并行 (软件方法)

- Cache 缺失的原因: 4C

6. 线程级并行

- Compulsory Misses (强制缺失): sad facts of life. Example: cold start misses.

7. 数据级并行 (向量处理)

- Capacity Misses (容量缺失): increase cache size

- Conflict Misses (冲突缺失): increase cache size and/or associativity

- Coherence (一致性缺失): Misses caused by cache coherence

8. 存储系统

9. 互连网络

- Cache性能

- 降低缺失率,

- 减少缺失代价

- 减少 Cache 命中时间



作业题

Write allocate

No-write allocate

有一个 4-way 组相联 Cache，Cache 大小为 16K 字节（4K 字），Cache 块大小为 16 字节。执行以下程序：

```
int M[4096], i, j;  
for (i = 0; i < 10; i++) {  
    for (j = 0; j < 4096; j++) {  
        M[j] = i + j;  
    }  
}
```

- 1、若 Cache 采用写分配（Write allocate）策略，计算执行整个程序时的 Cache 缺失率。
- 2、若 Cache 采用写不分配（No-write allocate）策略，计算执行整个程序时的 Cache 缺失率。
- 3、针对写不分配的 Cache 缺失率高的问题，试对程序进行优化，以取得最小的 Cache 缺失率。给出优化后的程序代码，并计算优化后的 Cache 缺失率。



解答：

1 若Cache采用写分配 (Write allocate) 策略，计算执行整个程序时的Cache 缺失率。

$$1024/4096*10 = 2.5\%$$

2 若Cache采用写不分配 (No-write allocate) 策略，计算执行整个程序时的Cache 缺失率。

$$100\%$$

3 针对写不分配的Cache缺失率高的问题，试对程序进行优化，以取得最小的Cache缺失率。给出优化后的程序代码，并计算优化后的Cache缺失率。

(1) 循环之前先读一遍 缺失率 $1024 / (4096 * 11) = 1/44$

```
for (i = 0 ; i < 4096 ; i++) {
```

```
    t=M[i]; }
```

(2) 循环之前先间隔为4读一遍 缺失率 $1024 / (4096 * 10 + 1024) = 1/41$

(3) 在循环内读一遍

```
for (i = 0 ; i < 10 ; i++) {  
    for (j = 0 ; j < 4096 ; j++) {  
        t=M[j];  
        M[ j ] = i + j;  
    }  
}
```

缺失率 $1024 / (4096 * 10 * 2) = 1/80$

答案主要有以下三种，其他还有一些，如读1次写2次的，只要先读就判对。

1. 基本概念&性能评价

1、**互连函数**：恒等函数、交换函数、均匀洗牌函数、逆均匀洗牌

2. 指令系统设计？

函数、碟式函数、反位序函数、移数函数、PM2I函数等

3. 流水线技术基础

2、**互连网络的结构参数与性能指标**：**网络规模、结点度、网络直径**、等分宽度、通信时延、网络时延、端口带宽、等分带宽等

4. 指令级并行（硬件方法）

3、**静态互连网络**：线性阵列、环和带弦环、全连接网络、循环移数网络、树形和星形、胖树形、网格形和环网形、超立方体等

5. 指令级并行（软件方法）

6. 线程级并行

4、**动态互连网络**：总线、开关、级控制、单元控制、部分级控制、交换等

7. 数据级并行（向量处理）

8. 存储系统

5、**消息传递机制**：路由、消息、包、寻径方式、线路交换、存储转发、虚拟直通、虫蚀方式、避免死锁、包冲突解决等。

9. 互连网络