计算机系统结构

课程总结（音频版）

谢长生

E-mail:cs_xie@mail.hust.edu.cn

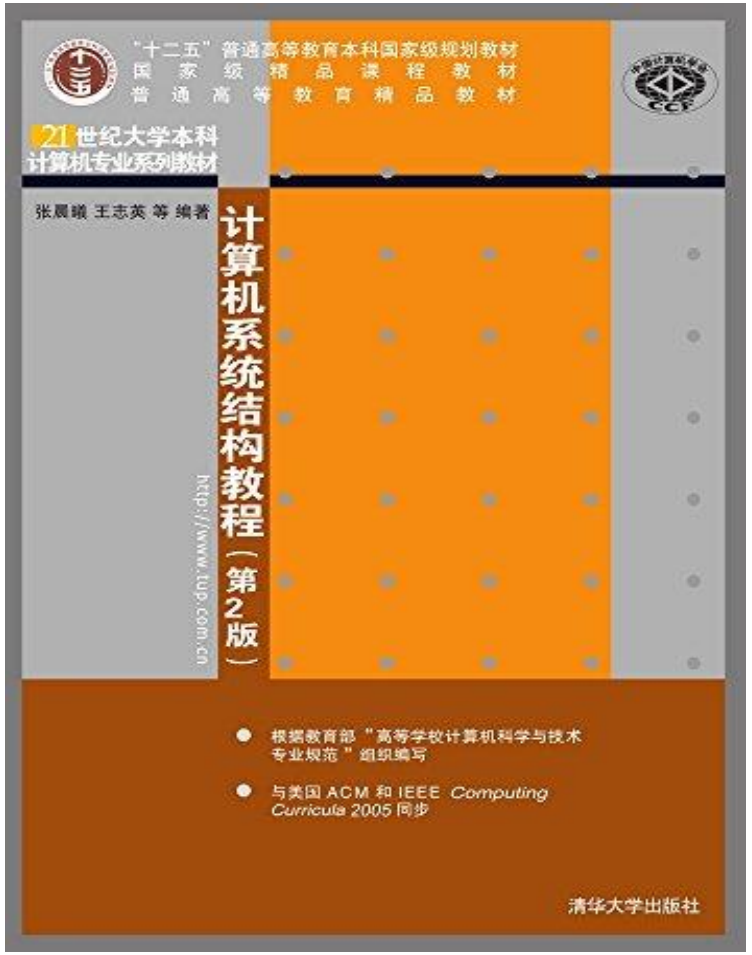
手机及微信：13871375398， QQ：1020984162

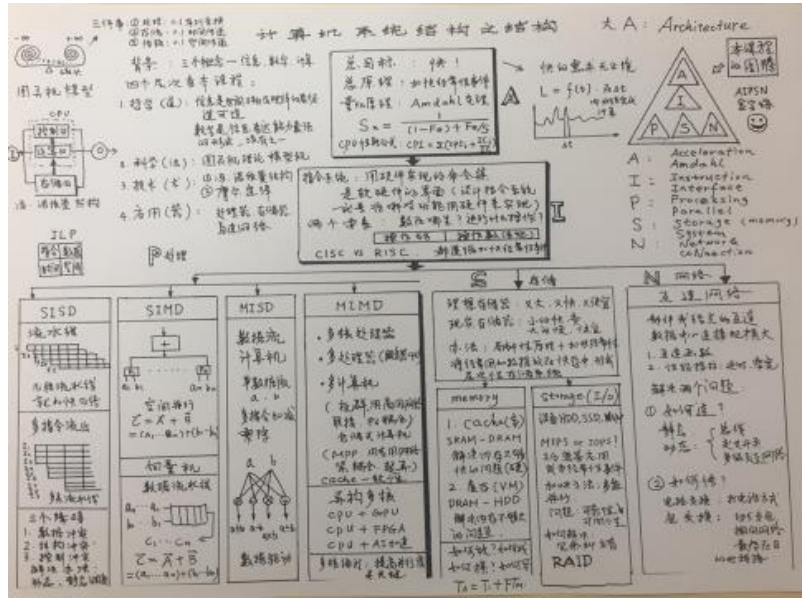
华中科技大学计算机学院

武汉光电国家研究中心光电信息存储研究部

读懂一本书，最好把它变为一页纸

——钱学森





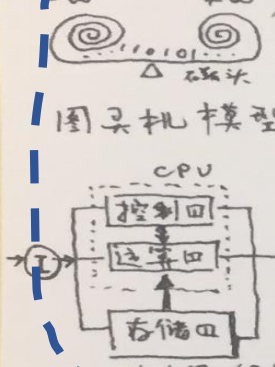
计算机系统结构之结构

大A：Architecture

三件事：① 处理：0.1 序列变换
② 存储：0.1 时间传递
③ 传输：0.1 空间传递

背景：三个概念——信息、数字、计算

图灵机模型



冯·诺依曼结构

四个层次看本课程：

1. 哲学（道）：信息是世间万物反规律的表征
道可道
数字是信息表达能力最强的形式，没有之一
2. 科学（法）：图灵机理论 模型机
3. 技术（术）：① 冯·诺依曼结构 ② 摩尔定律
4. 应用（器）：处理器、存储器、互连网络

总目标：快！

总原理：加快经常性事件

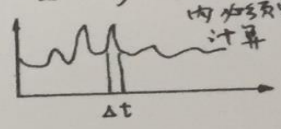
量化原理：Amdahl定理

$$S_n = \frac{1}{(1-Fe) + Fe/S}$$

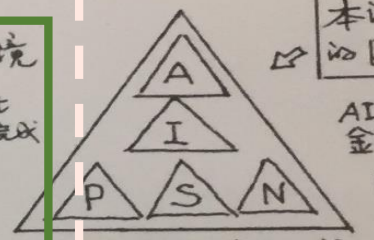
CPU性能公式：$CPI = \Sigma(CPI_i + \frac{IC_i}{TC})$

A → 快的要求无止境

$L = f(t)$，在 Δt 内必须完成计算



本课程的图腾



AIPSN 金字塔

A：Acceleration Amdahl
I：Instruction Interface
P：Processing Parallel
S：Storage (memory) System
N：Network Connection

指令系统：用硬件实现的命令集
是软硬件的界面（设计指令系统就是将哪些功能用硬件来实现）
两个要素：数在哪里？进行什么操作？

操作码	操作数（寻址）

CISC vs RISC：都遵循加快经常性事件

I

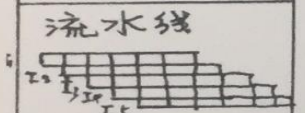
ILP

指令	数据
时间	空间

P 处理

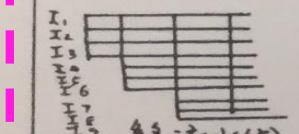
SISD

流水线



几级流水线就加快几倍

多指令流出

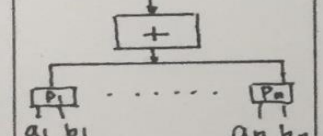


多条流水线

三个障碍
1. 数据冲突
2. 结构冲突
3. 控制冲突

解决办法：
动态、静态调度

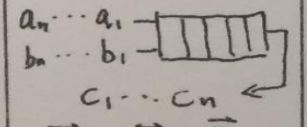
SIMD



空间并行
$\vec{C} = \vec{A} + \vec{B}$
$= (a_1 \cdots a_n) + (b_1 \cdots b_n)$

向量机

数据流水线



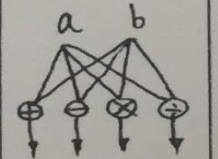
$\vec{C} = \vec{A} + \vec{B}$
$= (a_1 \cdots a_n) + (b_1 \cdots b_n)$

MISD

数据流计算机

单数据流 a、b

多指令加减乘除



数据驱动

MIMD

- 多核处理器
- 多处理器（服务器CPU）
- 多计算机

（机群，用商用网络联接，松耦合）
包储式计算机
（MPP，用专用网络紧耦合，超算）
cache一致性

异构多核
CPU + GPU
CPU + FPGA
CPU + AI加速

多核编程：提高并行度是关键

S 存储

理想存储器：又大、又快、又便宜

现实存储器：小的快、贵、大的慢、便宜

办法：局部性原理 + 加快经常性事件
将经常用的数据放在快存中，形成层次化存储系统

memory

1. cache($)
SRAM - DRAM
解决内存不够快的问题（硬）
2. 虚存（VM）
DRAM - HDD
解决内存不够大的问题。

如何放？如何找？
如何换？如何写？

$T_A = T_1 + FT_2$

Storage（I/O）

设备 HDD, SSD, NVM

MIPS or IOPS?

I/O密集应用成为经常性事件

加快方法：多盘并行

问题：可靠性↓ 可用性↓

如何解决：冗余+热备

RAID

N 网络

互连网络

部件或结点的互连
数据中心连接规模大

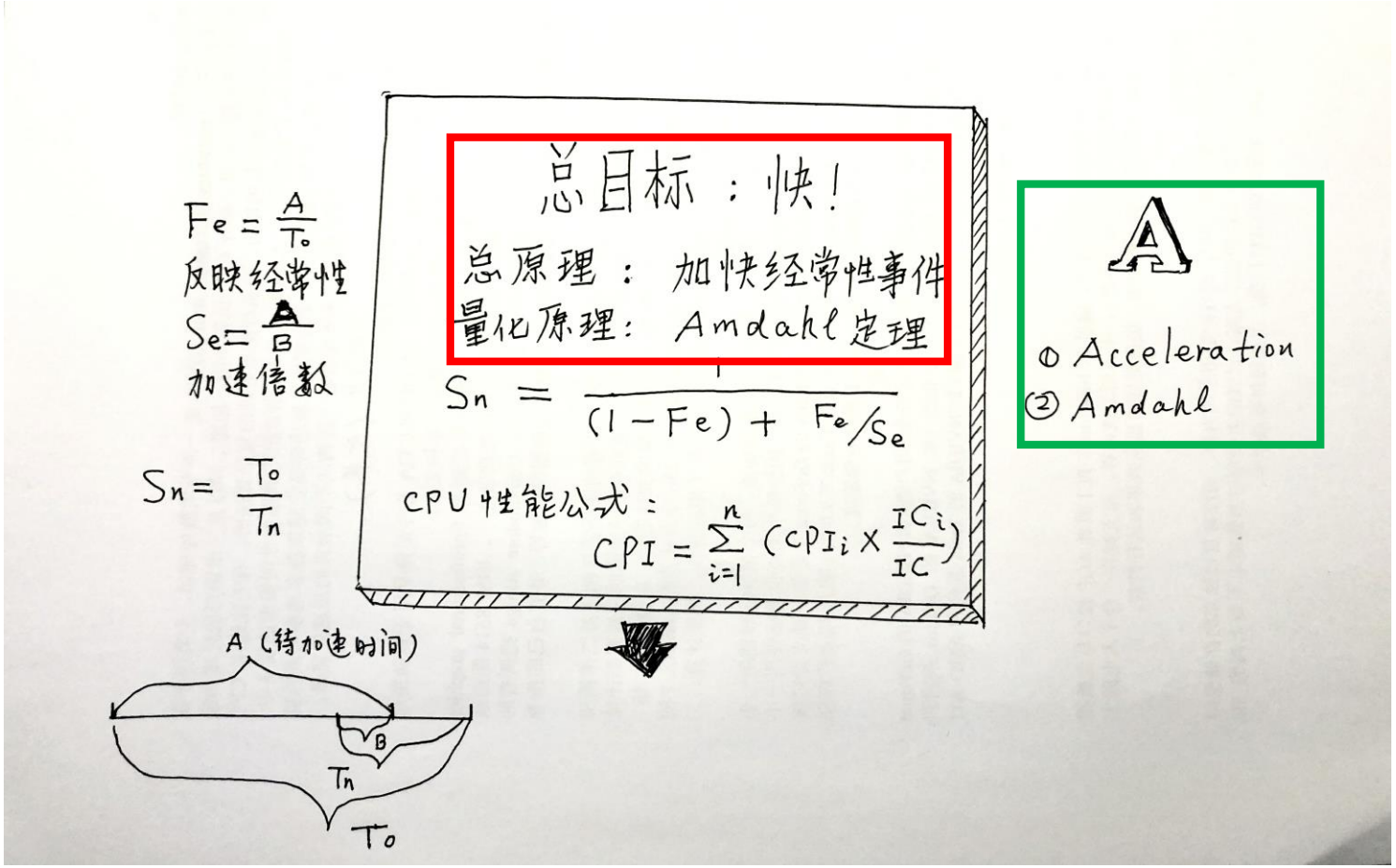
1. 互连函数
2. 性能指标：延时、带宽

解决两个问题：

① 如何连？
静态：总线、交叉开关、多级互连网络
动态：

② 如何传？
电路交换：打电话方式
包交换：切分为包，按网络最佳路径目的地拼接

第一框：系统结构总论(A)


总目标：快！
总原理：加快经常性事件
量化原理：Amdahl定理
$S_n = \frac{1}{(1-Fe) + Fe/Se}$
$Fe = \frac{A}{T_o}$
反映经常性
$Se = \frac{A}{B}$
加速倍数
$S_n = \frac{T_o}{T_n}$
CPU性能公式：
$CPI = \sum_{i=1}^{n} (CPI_i \times \frac{IC_i}{IC})$
A（待加速时间）
B
T_n
T_o
A
① Acceleration
② Amdahl

计算机系统结构之结构

大A：Architecture

三件事：① 处理：0-1序列变换
② 存储：0-1 时间传递
③ 传输：0-1 空间传递

图灵机模型

冯·诺依曼架构

背景：三个概念——信息、数字、计算

四个层次看本课程：

1. 哲学（道）：信息是世间万物及规律的表征，道可道
   数字是信息表达能力最强的形式，没有之一
2. 科学（法）：图灵机理论模型机
3. 技术（术）：① 冯·诺依曼结构 ② 摩尔定律
4. 应用（器）：处理器、存储器、互连网络

总目标：快！
总原理：加快经常性事件
量化原理：Amdahl定理

$$S_n = \frac{1}{(1-Fe) + Fe/S}$$

CPU性能公式：$CPI = \Sigma(CPI_i + \frac{IC_i}{TC})$

A

快的要求无止境

$L = f(t)$，在Δt内必须完成计算

本课程的图腾

AIPSN金字塔

A：Acceleration Amdahl
I：Instruction Interface
P：Processing Parallel
S：Storage (memory) System
N：Network coNnection

I

指令系统：用硬件实现的命令集，是软硬件的界面（设计指令系统就是将哪些功能用硬件来实现）

两个要素：数在哪里？进行什么操作？

操作码	操作数（寻址）

CISC vs RISC：都遵循加快经常性事件

ILP

指令	数据
时间	空间

P 处理

SISD

流水线

几级流水线就加快几倍

多指令流出

多条流水线

三个障碍
1. 数据冲突
2. 结构冲突
3. 控制冲突

解决办法：动态、静态调度

SIMD

空间并行

$\vec{C} = \vec{A} + \vec{B}$

$= (a_1 \cdots a_n) + (b_1 \cdots b_n)$

向量机

数据流水线

$\vec{C} = \vec{A} + \vec{B}$

$= (a_1 \cdots a_n) + (b_1 \cdots b_n)$

MISD

数据流计算机

单数据流 a、b

多指令加减乘除

数据驱动

MIMD

- 多核处理器
- 多处理器（服务器CPU）
- 多计算机

（机群，用商用网络联接，松耦合）

包储式计算机

（MPP，用专用网络紧耦合，超算）

cache一致性

异构多核
CPU + GPU
CPU + FPGA
CPU + AI加速

多核编程：提高并行度是关键

S 存储

理想存储器：又大、又快、又便宜

现实存储器：小的快、贵、大的慢、便宜

办法：局部性原理 + 加快经常性事件

将经常用的数据放在快存中，形成层次化存储系统

memory

1. cache（$）
SRAM - DRAM
解决内存不够快的问题（硬）
2. 虚存（VM）
DRAM - HDD
解决内存不够大的问题。

如何放？如何找？
如何换？如何写？

$T_A = T_1 + FT_M$

Storage（I/O）

设备HDD, SSD, NVM

MIPS or IOPS?

I/O密集应用成为经常性事件

加快方法：多盘并行

问题：可靠性↓ 可用性↓

如何解决：冗余+热备

RAID

N 网络

互连网络

部件或结点的互连

数据中心连接规模大

1. 互连函数
2. 性能指标：延时、带宽

解决两个问题：

① 如何连？

静态：总线 交叉开关 多级互连网络

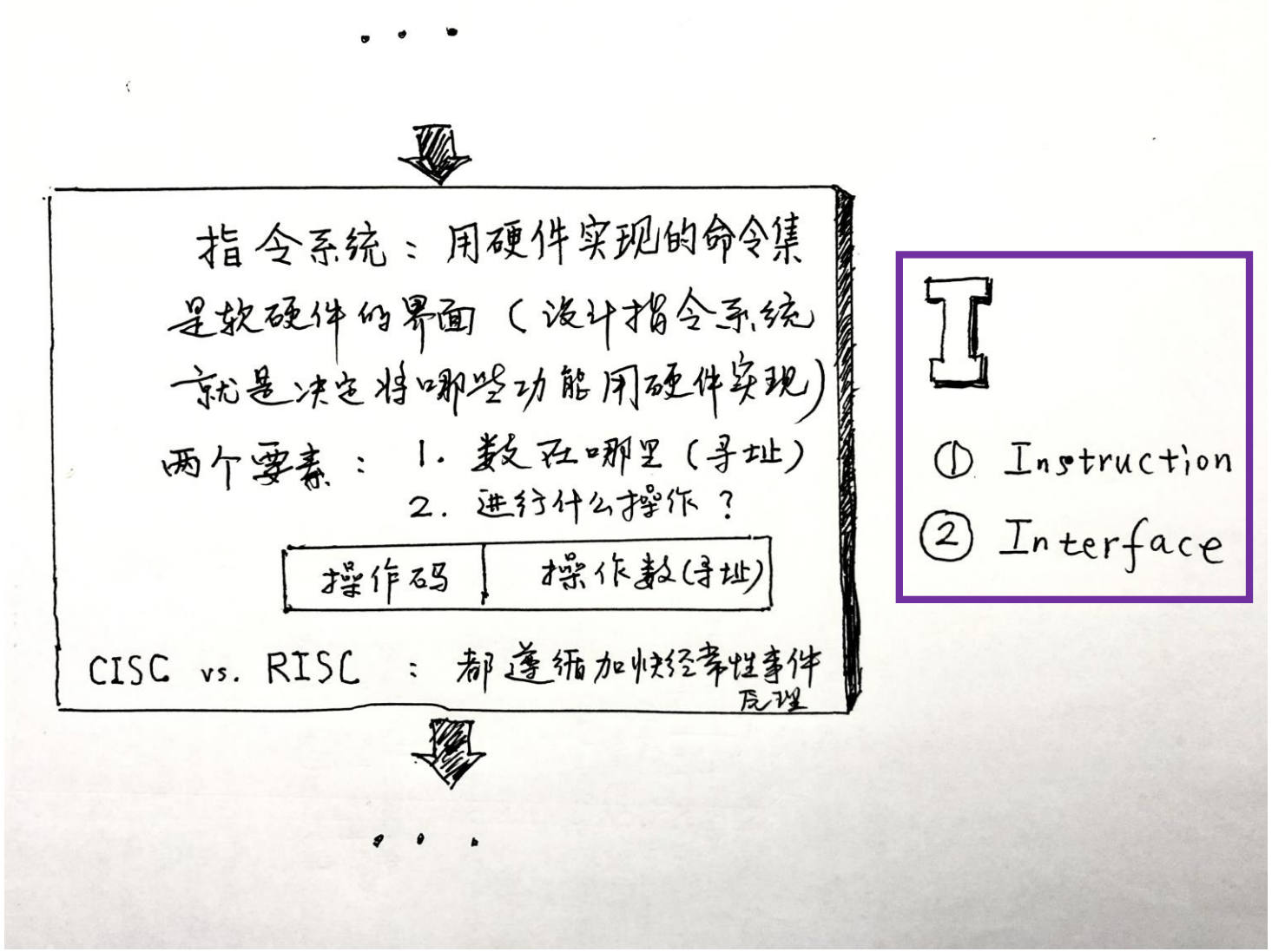
动态：

② 如何传？

电路交换：打电话方式

包交换：切分为包，投递网络，最后在目的地拼接

第二框：指令系统(Instruction)



I

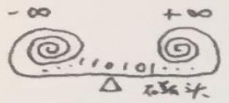
① Instruction

② Interface

计算机系统结构之结构

大A：Architecture

三件事：① 处理：0.1 序列变换
② 存储：0.1 时间传递
③ 传输：0.1 空间传递

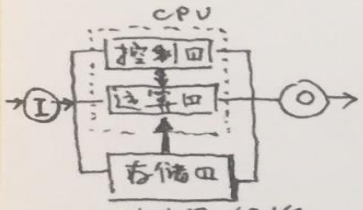


图灵机模型

背景：三个概念——信息、数字、计算

四个层次看本课程：

1. 哲学（道）：信息是世间万物及规律的表征
道可道
数字是信息表达能力最强的形式，没有之一
2. 科学（法）：图灵机理论模型机
3. 技术（术）：① 冯·诺依曼结构
② 摩尔定律
4. 应用（器）：处理器、存储器、互连网络



冯·诺依曼架构

总目标：快！
总原理：加快经常性事件
量化原理：Amdahl定理

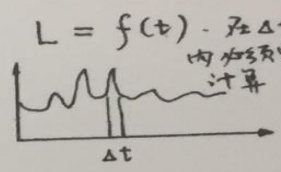
$$S_n = \frac{1}{(1-Fe) + Fe/S}$$

CPU性能公式：$CPI = \Sigma(CPI_i + \frac{IC_i}{TC})$

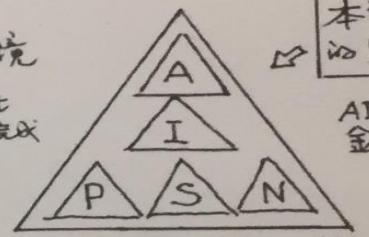
A

快的要求无止境

$L = f(t)$，在Δt内必须完成计算



本课程的图腾



AIPSN金字塔

A：Acceleration Amdahl
I：Instruction Interface
P：Processing Parallel
S：Storage (memory) System
N：Network coNnection

I

指令系统：用硬件实现的命令集
是软硬件的界面（设计指令系统就是将哪些功能用硬件来实现）
两个要素：数在哪里？进行什么操作？

操作码	操作数（寻址）

CISC vs RISC

都遵循加快经常性事件

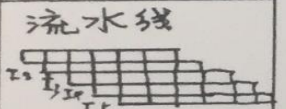
ILP

指令	数据
时间	空间

P 处理

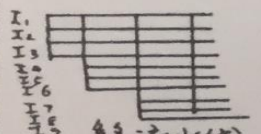
SISD

流水线



几级流水线就加快几倍

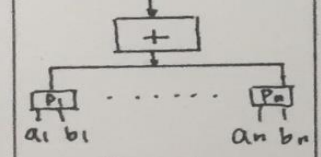
多指令流出



多条流水线

三个障碍
1. 数据冲突
2. 结构冲突
3. 控制冲突
解决办法：
动态、静态调度

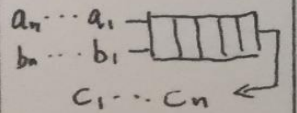
SIMD



空间并行
$\vec{C} = \vec{A} + \vec{B}$
$= (a_1 \cdots a_n) + (b_1 \cdots b_n)$

向量机

数据流水线



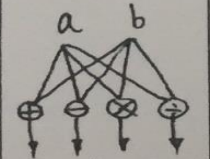
$\vec{C} = \vec{A} + \vec{B}$
$= (a_1 \cdots a_n) + (b_1 \cdots b_n)$

MISD

数据流计算机

单数据流
a、b

多指令加减乘除



数据驱动

MIMD

- 多核处理器
- 多处理器（服务器CPU）
- 多计算机

（机群，用商用网络联接，松耦合）
包储式计算机
（MPP，用专用网络紧耦合，超算）
cache一致性

异构多核

CPU + GPU
CPU + FPGA
CPU + AI加速

多核编程：提高并行度是关键

S 存储

理想存储器：又大、又快、又便宜
现实存储器：小的快、贵、大的慢、便宜
办法：局部性原理 + 加快经常性事件
将经常用的数据放在快存中，形成层次化存储系统

memory

1. cache($)
SRAM - DRAM
解决内存不够快的问题（硬）
2. 虚存（VM）
DRAM - HDD
解决内存不够大的问题。

如何放？如何找？
如何换？如何写？

$T_A = T_1 + FT_M$

Storage（I/O）

设备HDD, SSD, NVM

MIPS or IOPS?
I/O密集应用成为经常性事件
加快方法：多盘并行
问题：可靠性↓ 可用性↓
如何解决：冗余+纠检
RAID

N 网络

互连网络

部件或结点的互连
数据中心连接规模大

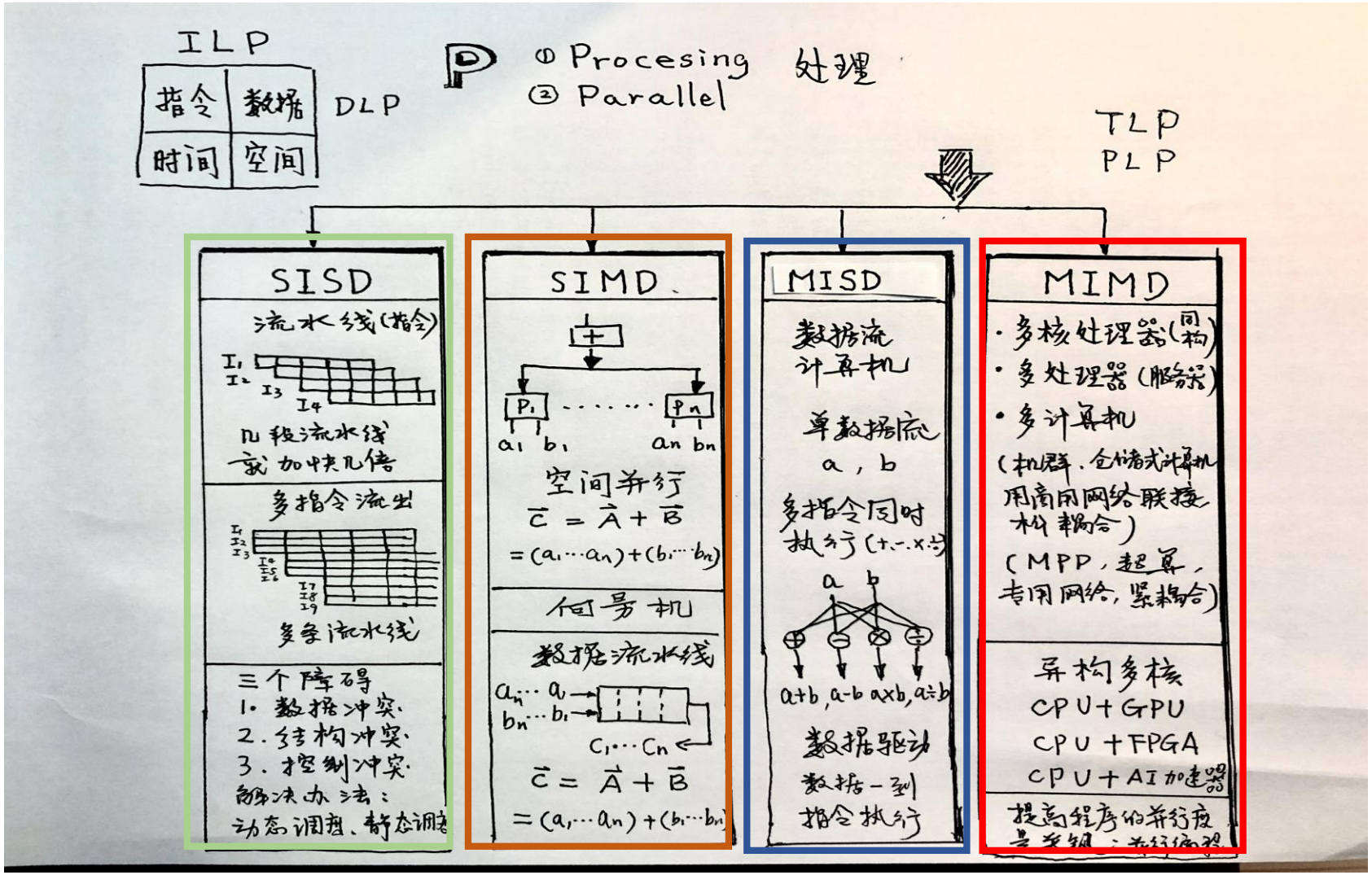
1. 互连函数
2. 性能指标：延时、带宽

解决两个问题：

① 如何连？
静态：总线、交叉开关、多级互连网络
动态：

② 如何传？
电路交换：打电话方式
包交换：切分为包，投递网络，最后在目的地拼接

第三框：处理（Processing）

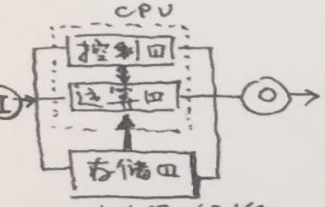


计算机系统结构之结构

大A：Architecture

三件事：① 处理：0-1序列变换
② 存储：0-1 时间传递
③ 传输：0-1 空间传递

图灵机模型



冯·诺依曼架构

背景：三个概念——信息、数字、计算

四个层次看本课程：

1. 哲学（道）：信息是世间万物及规律的表征
道可道
数字是信息表达能力最强的形式，没有之一
2. 科学（法）：图灵机理论模型机
3. 技术（术）：① 冯·诺依曼结构
② 摩尔定律
4. 应用（器）：处理器、存储器、互连网络

总目标：快！
总原理：加快经常性事件
量化原理：Amdahl定理

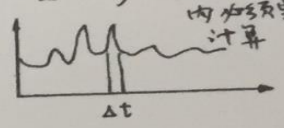
$$S_n = \frac{1}{(1-Fe) + Fe/S}$$

CPU性能公式：$CPI = \Sigma(CPI_i + \frac{IC_i}{TC})$

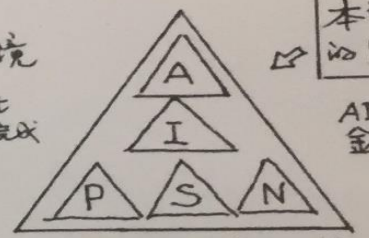
A

快的要求无止境

$L = f(t)$，在Δt内必须完成计算



本课程的图腾



AIPSN 金字塔

A：Acceleration Amdahl
I：Instruction Interface
P：Processing Parallel
S：Storage (memory) System
N：Network connection

I

指令系统：用硬件实现的命令集
是软硬件的界面（设计指令系统就是将哪些功能用硬件来实现）
两个要素：数在哪里？进行什么操作？

操作码	操作数（寻址）

CISC vs RISC：都遵循加快经常性事件

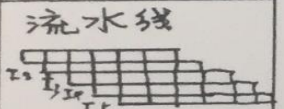
ILP

指令	数据
时间	空间

P 处理

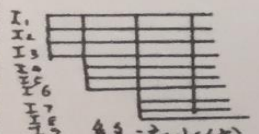
SISD

流水线



几级流水线就加快几倍

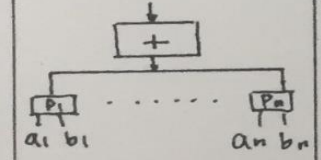
多指令流出



多条流水线

三个障碍
1. 数据冲突
2. 结构冲突
3. 控制冲突
解决办法：动态、静态调度

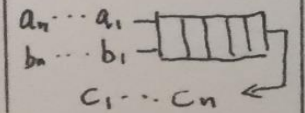
SIMD



空间并行
$\vec{C} = \vec{A} + \vec{B}$
$= (a_1 \cdots a_n) + (b_1 \cdots b_n)$

向量机

数据流水线



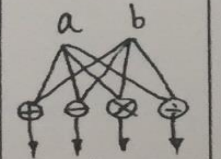
$\vec{C} = \vec{A} + \vec{B}$
$= (a_1 \cdots a_n) + (b_1 \cdots b_n)$

MISD

数据流计算机

单数据流
a、b

多指令加减乘除



数据驱动

MIMD

- 多核处理器
- 多处理器（服务器CPU）
- 多计算机

（机群，用商用网络联接，松耦合）
包储式计算机
（MPP，用专用网络紧耦合，超算）
cache一致性

异构多核
CPU + GPU
CPU + FPGA
CPU + AI加速

多核编程：提高并行度是关键

S 存储

理想存储器：又大、又快、又便宜
现实存储器：小的快、贵、大的慢、便宜
办法：局部性原理 + 加快经常性事件
将经常用的数据放在快存中，形成层次化存储系统

memory

1. cache($)
SRAM - DRAM
解决内存不够快的问题（硬）
2. 虚存（VM）
DRAM - HDD
解决内存不够大的问题。

如何放？如何找？
如何换？如何写？

$T_A = T_c + T_m$

Storage（I/O）

设备HDD, SSD, NVM

MIPS or IOPS?
I/O密集应用成为经常性事件
加快方法：多盘并行
问题：可靠性、可用性
如何解决：冗余+纠检
RAID

N 网络

互连网络

部件或结点的互连
数据中心连接规模大

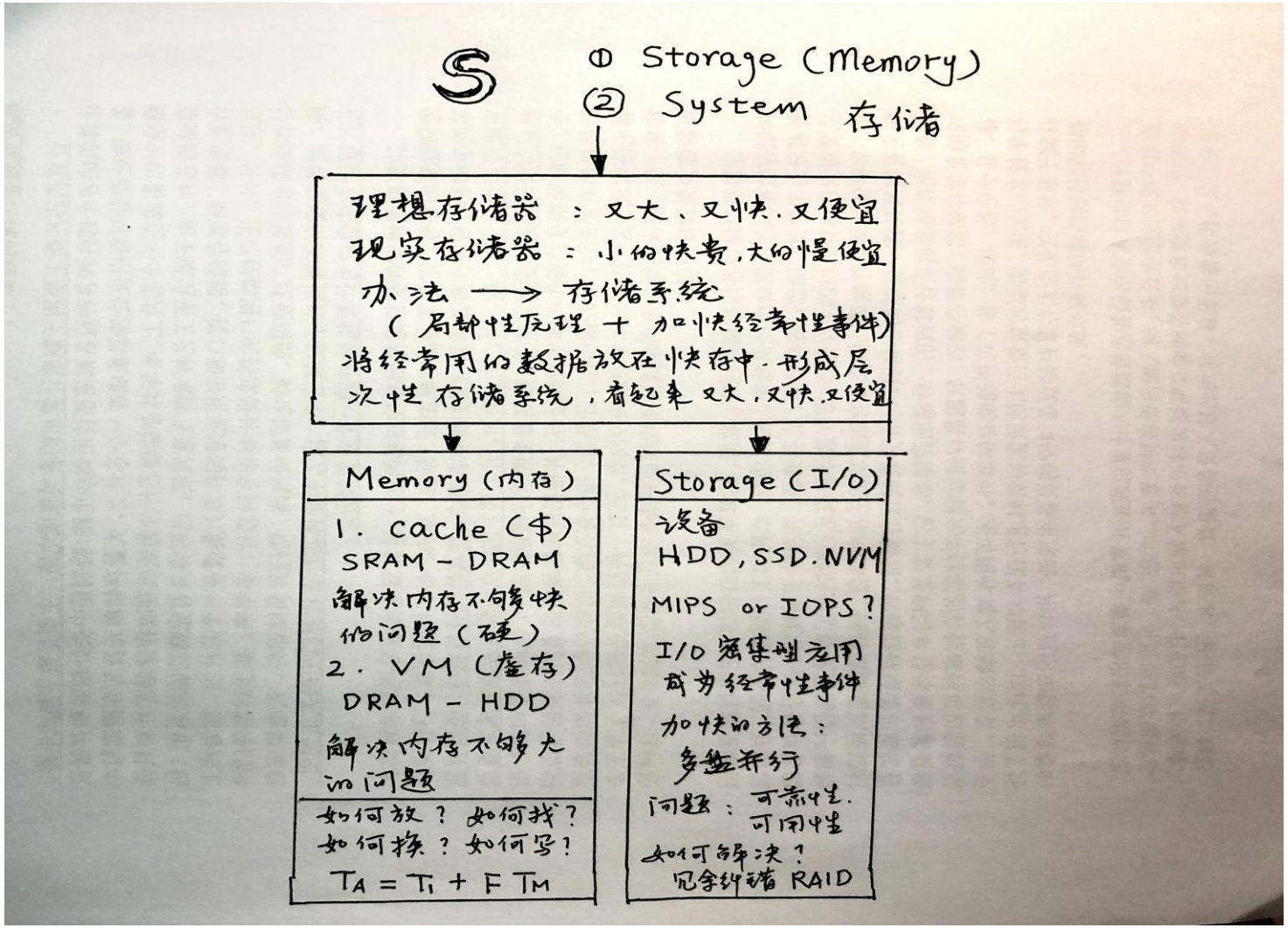
1. 互连函数
2. 性能指标：延时、带宽

解决两个问题：

① 如何连？
静态：总线、交叉开关、多级互连网络
动态：

② 如何传？
电路交换：打电话方式
包交换：切分为包，报文网络，最后在目的地拼接

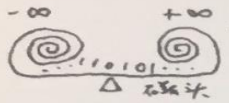
第四框：存储（Storage,memory)



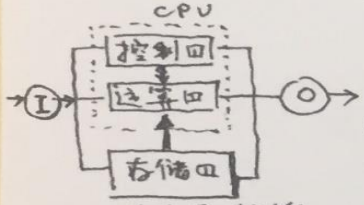
计算机系统结构之结构

大A：Architecture

三件事：① 处理：0.1 序列变换
② 存储：0.1 时间传递
③ 传输：0.1 空间传递



图灵机模型



冯·诺依曼结构

ILP

指令	数据
时间	空间

背景：三个概念——信息、数字、计算

四个层次看本课程：

1. 哲学（道）：信息是世间万物反规律的表征
道可道
数字是信息表达能力最强的形式，没有之一
2. 科学（法）：图灵机理论模型机
3. 技术（术）：① 冯·诺依曼结构
② 摩尔定律
4. 应用（器）：处理器、存储器、互连网络

P 处理

总目标：快！
总原理：加快经常性事件
量化原理：Amdahl定理

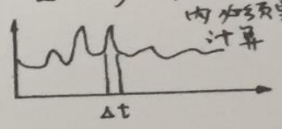
$$S_n = \frac{1}{(1-Fe)+Fe/S}$$

CPU性能公式：$CPI = \Sigma(CPI_i + \frac{IC_i}{TC})$

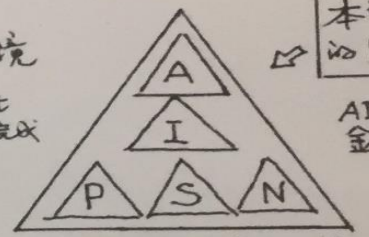
A

快的需求无止境

$L = f(t)$，在Δt内必须完成计算



本课程的图腾



AIPSN 金字塔

A：Acceleration Amdahl
I：Instruction Interface
P：Processing Parallel
S：Storage (memory) System
N：Network connection

指令系统：用硬件实现的命令集
是软硬件的界面（设计指令系统就是将哪些功能用硬件来实现）
两个要素：数在哪里？进行什么操作？

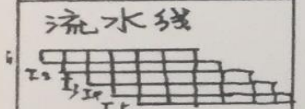
操作码	操作数（寻址）

CISC vs RISC：都遵循加快经常性事件

I

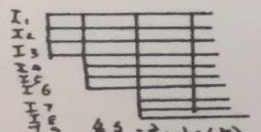
SISD

流水线



几级流水线就加快几倍

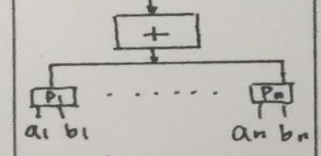
多指令流出



多条流水线

三个障碍
1. 数据冲突
2. 结构冲突
3. 控制冲突
解决办法：动态、静态调度

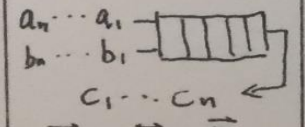
SIMD



空间并行
$\vec{C} = \vec{A} + \vec{B}$
$= (a_1 \cdots a_n) + (b_1 \cdots b_n)$

向量机

数据流水线



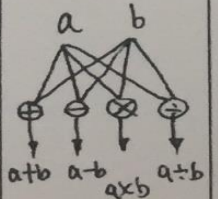
$\vec{C} = \vec{A} + \vec{B}$
$= (a_1 \cdots a_n) + (b_1 \cdots b_n)$

MISD

数据流计算机

单数据流
a、b

多指令加减乘除



数据驱动

MIMD

- 多核处理器
- 多处理器（服务器CPU）
- 多计算机

（机群，用商用网络联接，松耦合）
包储式计算机
（MPP，用专用网络紧耦合，超算）
cache一致性

异构多核
CPU + GPU
CPU + FPGA
CPU + AI加速

多核编程：提高并行度是关键

S 存储

理想存储器：又大、又快、又便宜
现实存储器：小的快、贵、大的慢、便宜
办法：局部性原理 + 加快经常性事件
将经常用的数据放在快存中，形成层次化存储系统

memory

1. cache（$）
SRAM - DRAM
解决内存不够快的问题（硬）
2. 虚存（VM）
DRAM - HDD
解决内存不够大的问题。

如何放？如何找？
如何换？如何写？

$T_A = T_1 + FT_M$

Storage（I/O）

设备HDD, SSD, NVM

MIPS or IOPS?
I/O密集应用成为经常性事件
加快方法：多盘并行
问题：可靠性、可用性
如何解决：冗余+备份
RAID

N 网络

互连网络

部件或结点的互连
数据中心连接规模大

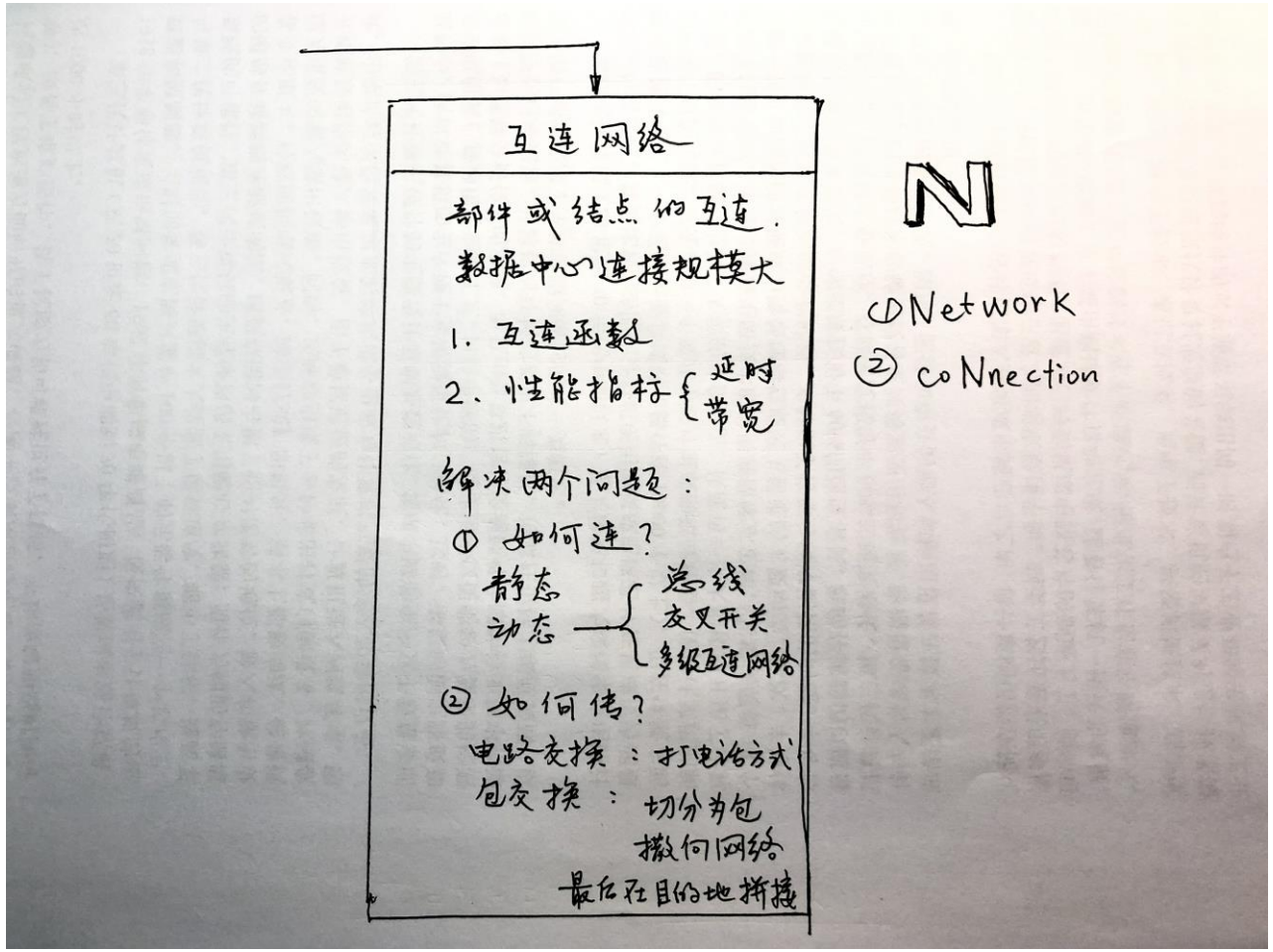
1. 互连函数
2. 性能指标：延时、带宽

解决两个问题：

① 如何连？
静态：总线、交叉开关、多级互连网络
动态：

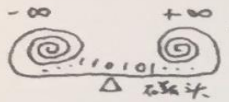
② 如何传？
电路交换：打电话方式
包交换：切分为包，按网络最佳路径拼接

第五框：互连网络（Network）

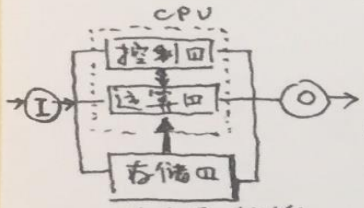


计算机系统结构之结构

三件事：① 处理：0-1序列变换
② 存储：0-1 时间传递
③ 传输：0-1 空间传递



图灵机模型



冯·诺依曼架构

ILP

指令	数据
时间	空间

背景：三个概念——信息、数字、计算

四个层次看本课程：

1. 哲学（道）：信息是世间万物及规律的表征
道可道
数字是信息表达能力最强的形式，没有之一
2. 科学（法）：图灵机理论模型机
3. 技术（术）：① 冯·诺依曼结构
② 摩尔定律
4. 应用（器）：处理器、存储器、互连网络

A

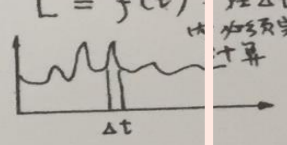
总目标：快！
总原理：加快经常性事件
量化原理：Amdahl定理

$$S_n = \frac{1}{(1-Fe) + Fe/S}$$

CPU性能公式：$CPI = \Sigma(CPI_i + \frac{IC_i}{TC})$

快的要求无止境

$L = f(t)$



在Δt必须完成计算

I

指令系统：用硬件实现的命令集
是软硬件的界面（设计指令系统就是将哪些功能用硬件来实现）
两个要素：数在哪里？进行什么操作？

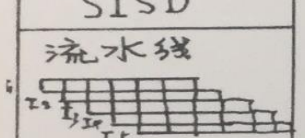
操作码	操作数（寻址）

CISC vs RISC：都遵循加快经常性事件

P 处理

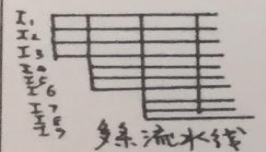
SISD

流水线



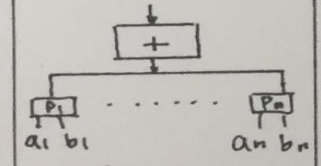
几级流水线就加快几倍

多指令流出



三个障碍
1. 数据冲突
2. 结构冲突
3. 控制冲突
解决办法：动态、静态调度

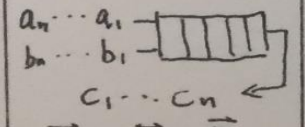
SIMD



空间并行
$\vec{C} = \vec{A} + \vec{B}$
$= (a_1 \cdots a_n) + (b_1 \cdots b_n)$

向量机

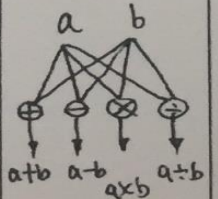
数据流水线



$\vec{C} = \vec{A} + \vec{B}$
$= (a_1 \cdots a_n) + (b_1 \cdots b_n)$

MISD

数据流计算机
单数据流
a、b
多指令加减乘除



数据驱动

MIMD

- 多核处理器
- 多处理器（服务器CPU）
- 多计算机

（机群，用商用网络联接，松耦合）
包储式计算机
（MPP，用专用网络紧，耦合，起算）
cache一致性

异构多核
CPU + GPU
CPU + FPGA
CPU + AI加速

多核编程：提高并行度是关键

S 存储

理想存储器：又大，又快，又便宜
现实存储器：小的快，贵
大的慢，便宜
办法：局部性原理 + 加快经常性事件
将经常用的数据放在快存中，形成层次化存储系统

memory

1. cache($)
SRAM - DRAM
解决内存不够快的问题（硬）
2. 虚存（VM）
DRAM - HDD
解决内存不够大的问题。

如何放？如何找
如何换？如何写

$T_A = T_1 + FT_M$

Storage（I/O）

设备HDD，SSD，NVM

MIPS or IOPS?
I/O密集应用成为经常性事件
加快方法：多盘并行
问题：可靠性↓
可用性↓
如何解决：冗余+备份
RAID

N 网络

互连网络

部件或结点的互连
数据中心连接规模大

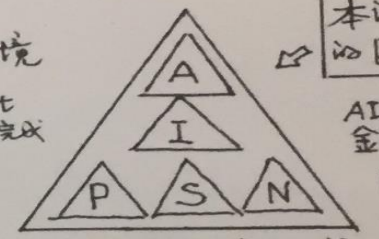
1. 互连函数
2. 性能指标：延时、带宽

解决两个问题：

① 如何连？
静态：总线、交叉开关、多级互连网络
动态：

② 如何传？
电路交换：打电话方式
包交换：切分为包，投递网络，最后在目的地拼接

大A：Architecture



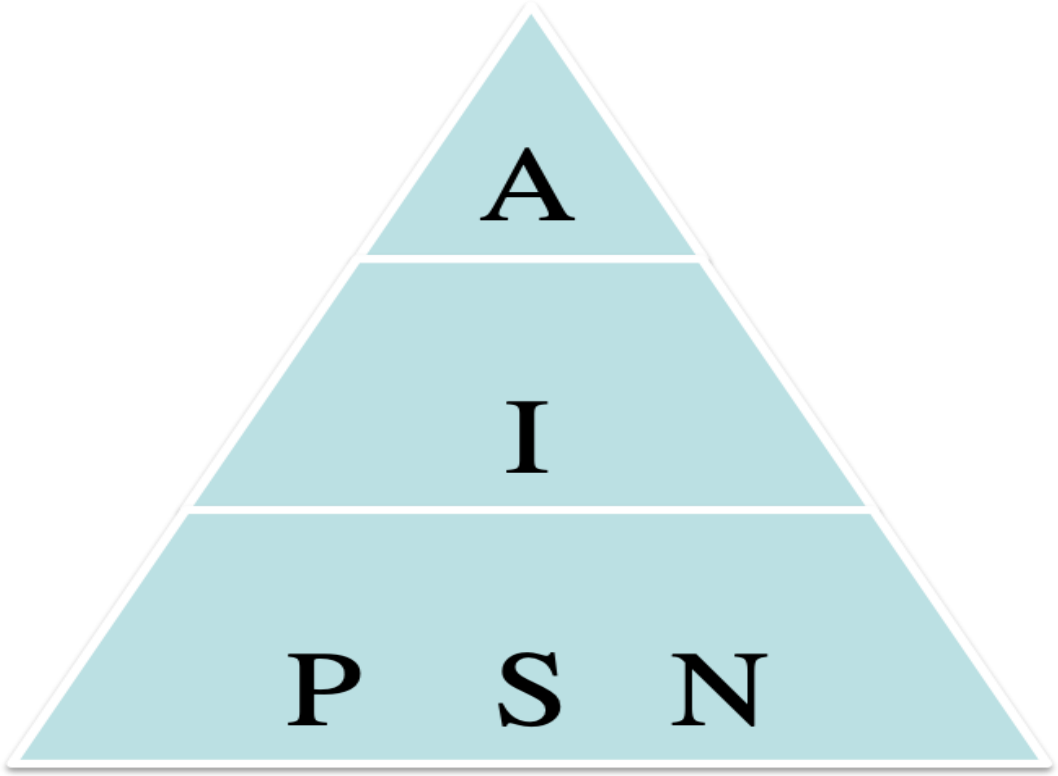
本课程的图腾

AIPSN 金字塔

A：Acceleration Amdahl
I：Instruction Interface
P：Processing Parallel
S：Storage (memory) System
N：Network coNnection

本课程的总“图腾”

- AIPSN金字塔



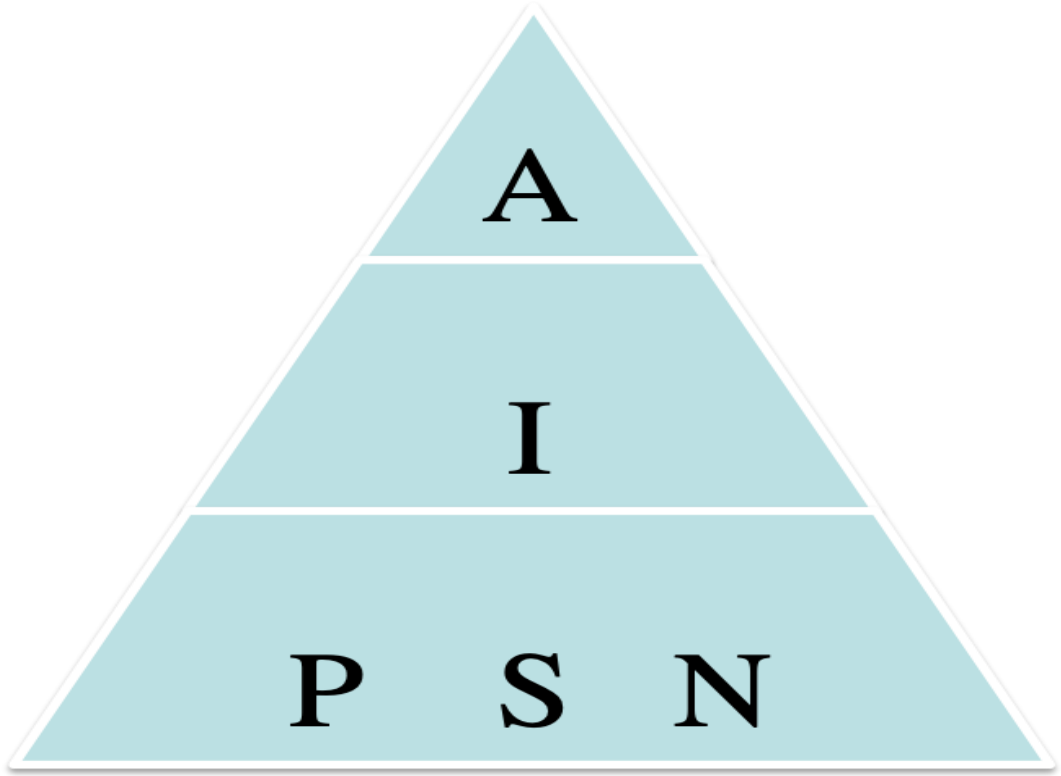
Architecture

- A：Acceleration、Amdahl
- I ：Instruction、Interface
- P：Processing、Parallel
- S ：Storage(Memory)、System
- N：Network、coNnection

课程一句话总结:

什么是计算机系统结构的全部内容?

在阿姆达定理的指导下，通过指令系统，对数字进行处理、存储、传输



全剧终

Thank You!

谢谢，再见！

背景音乐：Time to Say Goodbye