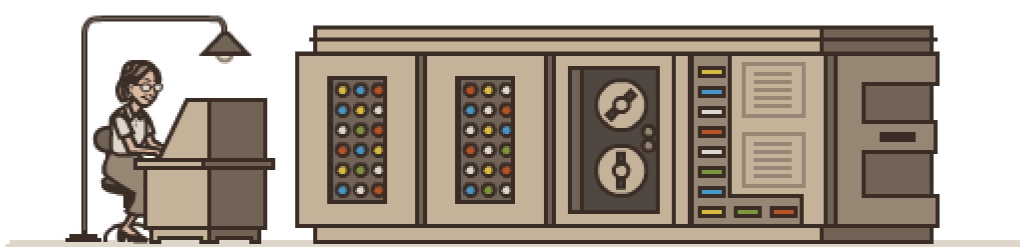
计算机组成原理
第一章 概论

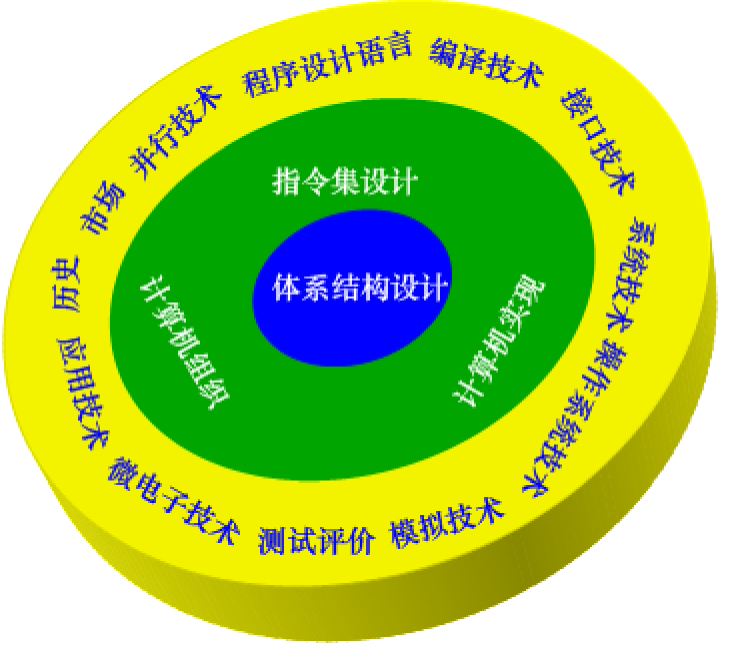
李曦

llxx@ustc.edu.cn



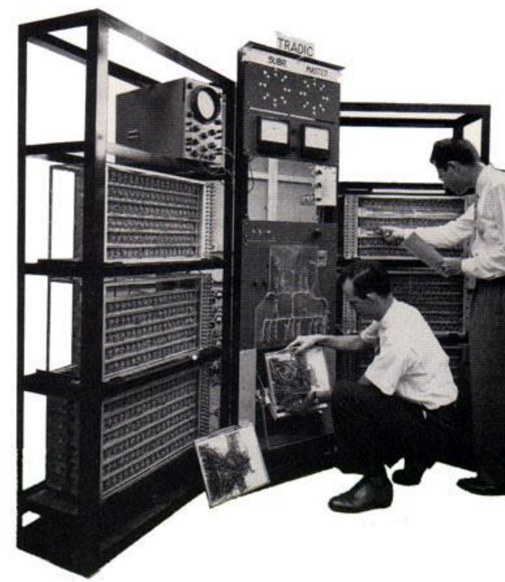
本章教学内容

- 计算机组成的科学基础及发展史
- 计算机系统概述
  - 计算机软硬件
  - 计算机系统的层次结构
  - 计算机组成 VS. 计算机体系结构
- 计算机的基本组成：AB
  - Von Neumann机的特征
  - 计算机的硬件构成
  - 计算机的工作过程
- 计算机硬件的性能指标：C
- 数电基础知识回顾



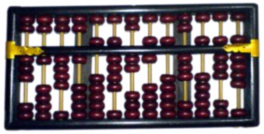
计算机系统研究的两条线

- 计算机理论：图灵
  - Turing Machine，1936
    - 可计算性，计算复杂性
    - 图灵问题：智能的机械化
  - 存储程序?

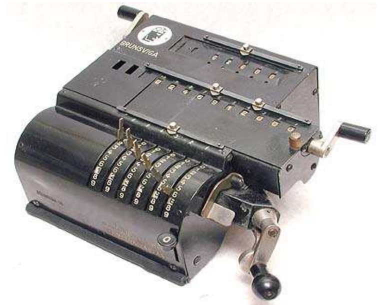


- 计算机工程：冯·诺依曼
  - von Neuman Machine，1945
  - 存储程序：$2.1@RV2
    - “指令和数据都存储在存储器中，易于更改”

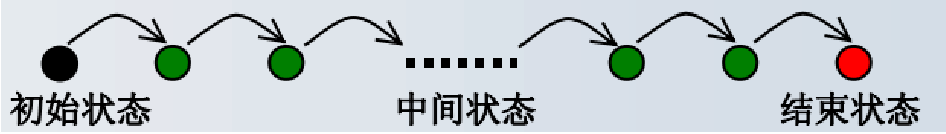
可计算性理论(computability theory)



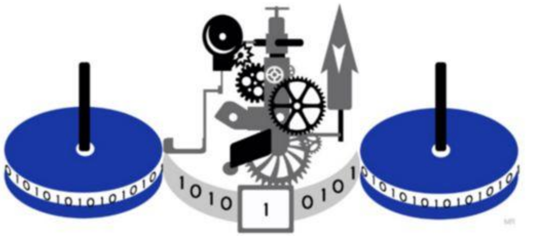
- 计算：是对运算过程的一种高度抽象
- 算法
  - 对计算的步骤或状态的一种刻画，是计算方法的一种实现方式
    - 将计算抽象为输入到输出的函数映射，是一个封闭的计算过程
  - 算法可计算性：判断一类数学问题是否机械可解
    - 可计算问题：算术逻辑运算，量化，离散
    - 非可计算问题：明天是否下雨？



- 计算模型（MoC）
  - 刻画“计算”概念的抽象的数学系统或形式化系统。
    - 数学系统：λ演算(串行、递归)、π演算（并行、分布）等
    - 形式化系统：状态迁移系统、自动机等
      - 具有状态转换特征，能够对所处理的对象的数据或信息进行表示、加工、变换、输出的数学机器。
      - 图灵机



MoC：图灵机(Turing Machine,1936)



- 自动计算机的结构与行为（ABC）
  - 一条两端可以无限延伸的纸带
  - 一个读写头：符号包括0、1、b
  - 一个控制器：执行控制读写头工作的命令
  - 顺序执行：状态机
    - 五元组“（状态、读符号）→（写符号、移动、状态）”
      - 状态集：开始态，中间态，结束态
      - 当进入结束态时，停机(H)
      - 六个操作原语（primitives）：读、写、左、右、擦除、停止

控制命令示例：

$q_1 01Rq_1$
$q_1 10Rq_1$
$q_1 bbRq_2$
$q_2 bbLq_3$
$q_2 00Hq_1$
$q_2 11Hq_1$

- 并行图灵机？
- 算盘=计算机？

关于MoC的两个重要原理

- 问题
  - 计算复杂性是否与计算模型有关?
  - 不同计算模型解决同一类问题所需资源是否相同?
- 相似性原理
  - 相似性原理：所有计算模型的计算能力等同
    - 所有合理的、功能足够强大的计算模型可以相互模拟计算，且使用的本质相同的并行计算时间、串行计算时间和空间
    - Turing完备性
  - 丘奇一图灵论题：可计算性等价于图灵机的可计算性
- 对偶性原理
  - 在并行计算模型上，计算的时间与空间可以互换

图灵完备性Turing-complete/Turing-equivalent

- 图灵机：六个基本原语【primitive or atomic action】
  - 读、写、擦除、左移、右移、停机
  - 图灵完备性系统：能够模拟图灵机
- 图灵完备语言
  - 最小图灵完备语言BF（1993）
    - 机器模型+8种运算符
- 非图灵完备语言
  - 如数据描述语言：HTML，XML…

Brainfuck	C
`>`	`++ptr;`
`<`	`--ptr;`
`+`	`++*ptr;`
`-`	`--*ptr;`
`.`	`putchar(*ptr);`
`,`	`*ptr =getchar();`
`[`	`while (*ptr) {`
`]`	`}`

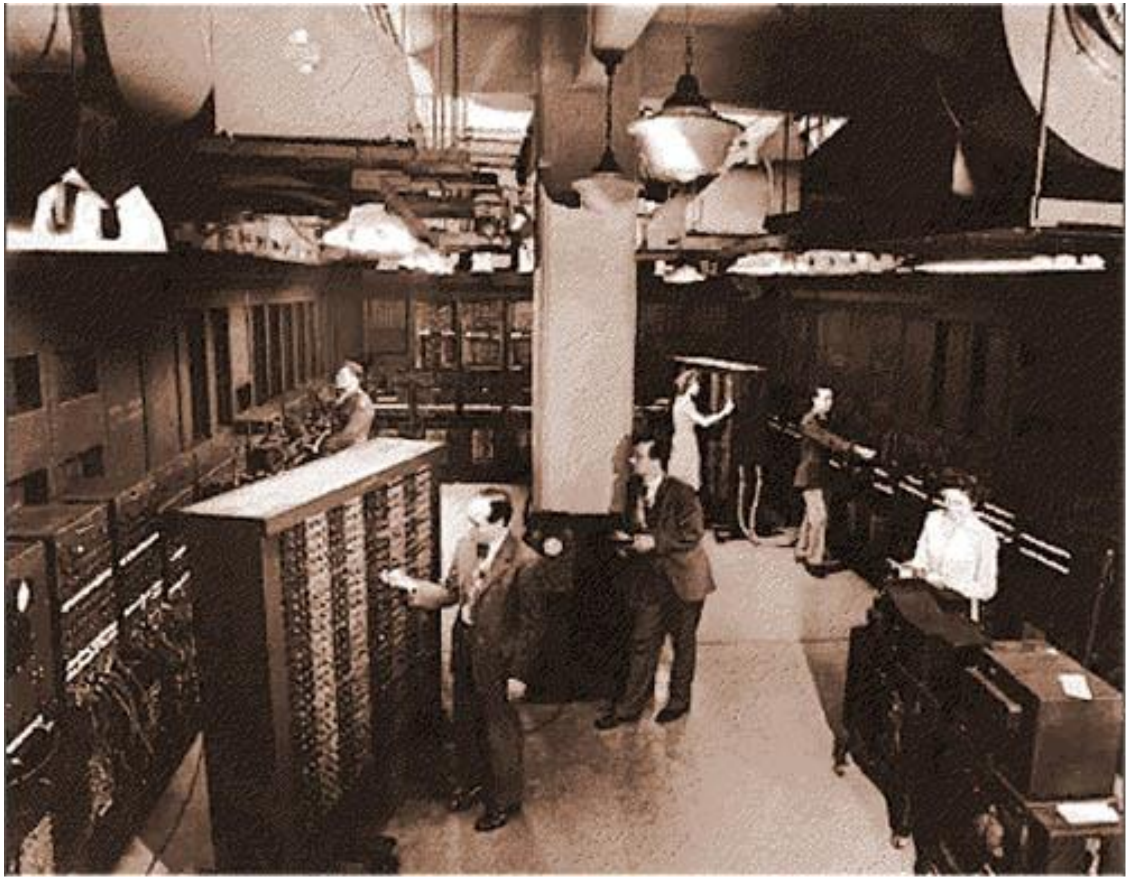
BF在屏幕上打印"Hello World!"

```
++++++++++[>+++++++>++++++++++>+++>+<<<<-]
>++.>+.+++++++..+++.>++.<<+++++++++++++++.
>.+++.------.--------.>+.>.
```

第一台现代电子计算机，真空管，1946公开

RV2：$1.13
FIGURE e1.13.1
二战期间研制
J. Presper Eckert
and John Mauchly主持，
1944年冯诺依曼参与，提出“存储程序”想法

唐$2.1



- ENIAC(Electronic Numerical Integrator and Computer)，1946年宾夕法尼亚大学
  - 运算速度 5000次/秒，功耗150kw/h，占地170m²，造价100万美元。用于测定氢弹可靠性。
- **不具备“存储程序”**能力——不可编程，且程序无法共享
  - 程序要通过外接电路板输入（wired）。对于不同类型的计算，需要设计相应的外接插板。
  - 十进制并行计算机（同时处理10个数）。

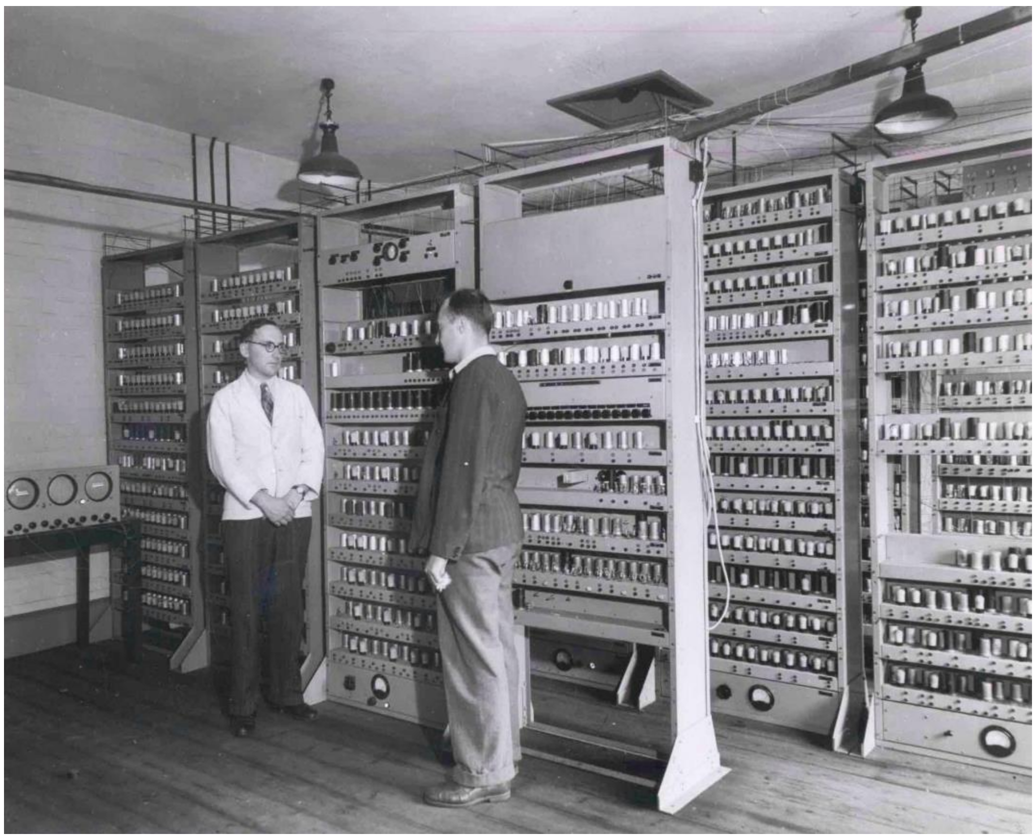
EDVAC计算机，1944～1952

- Electronic Discrete Variable Automatic Computer
  - 1MHz，二进制，字长32位，串行
  - 首次提出“存储程序(Stored Program)”

冯诺依曼（John von Neumann）和奥本海默（Julius Robert Oppenheimer）与普林斯顿计算机

第一台存储程序式计算机EDSAC

- Electronic Delay Storage Automatic Calculator
  - 参考EDVAC机，1946～1949年Wilkes在剑桥实现

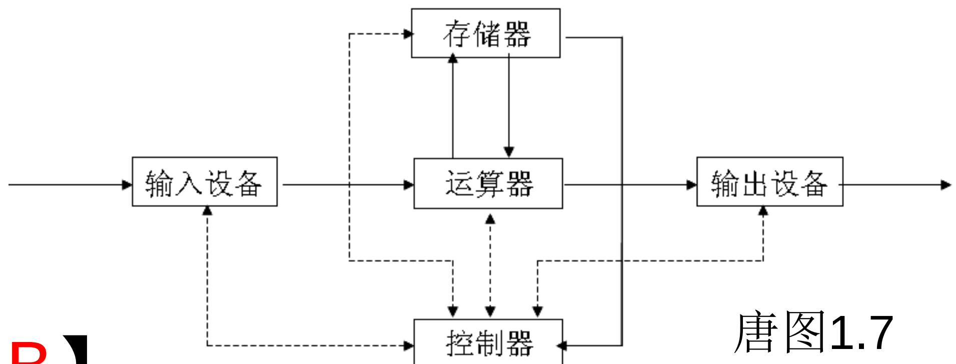


- 采用水银延迟线为存储器，可存储34b字长的512字。
- 加法时间1.5ms，乘法时间4ms。
- 串行计算机（数据传输和运算按位逐一进行）
- *微程序*，*子程序*，*Cache*，…

Maurice Vincent Wilkes，第二届图灵奖，1967

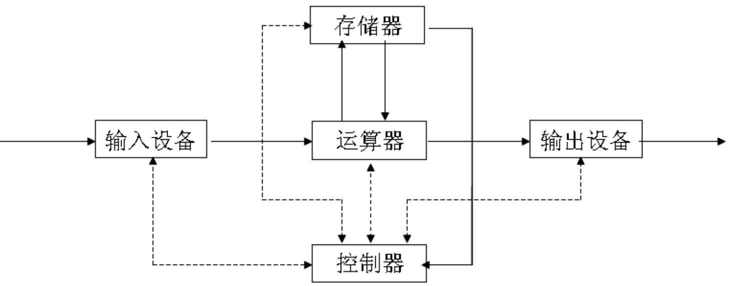
Von Neumann Machine Arch，ABC，唐$1.2.1

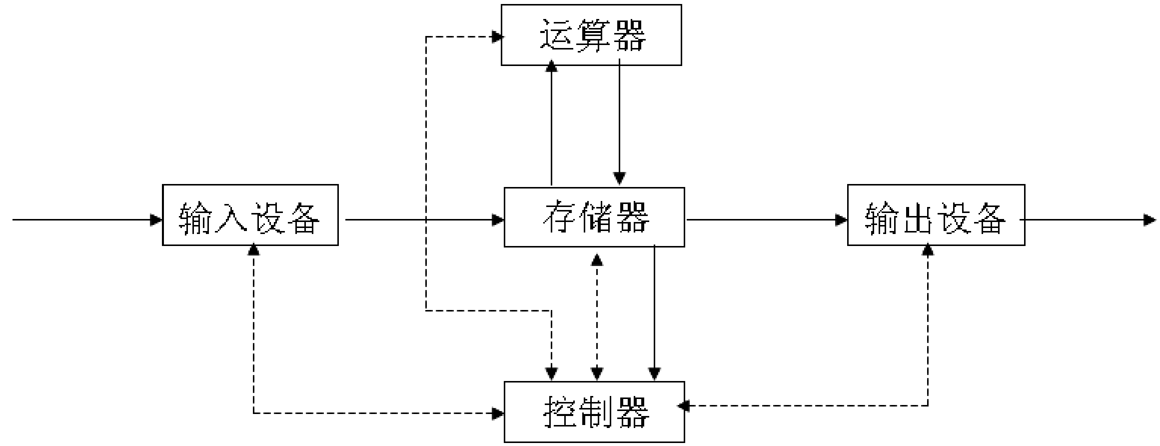
- “存储程序计算机”：由五大部件构成
  - 运算器、控制器、存储器、输入设备、输出设备；【互连？】
- “存储程序”：指令和数据存储方式
  - 指令和数据“以同等地位”存放于存储器，分别按地址访问
- 指令和数据表示形式
  - 均用二进制码表示
- 指令字组成
  - 由操作码和地址码构成
- 程序控制【命令式！】【B】
  - 存储程序式计算机：指令按**顺序**存放，**顺序**执行；【非并行】
- 数据传输
  - 机器以运算器为中心
- 特点：存储与计算相分离——亦称**普林斯顿结构**



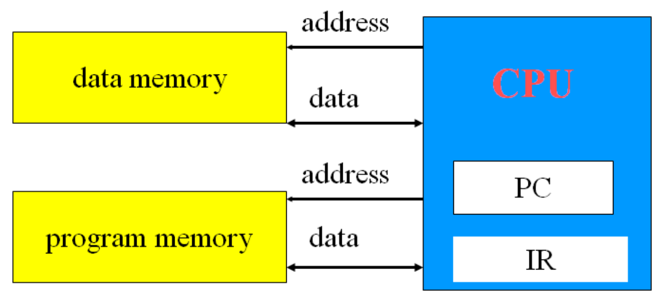
唐图1.7

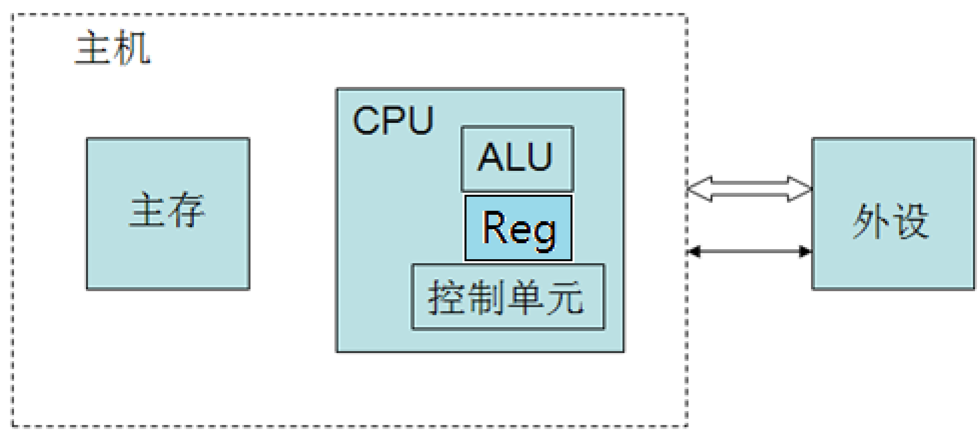
现代计算机的组织结构特征（ABC）





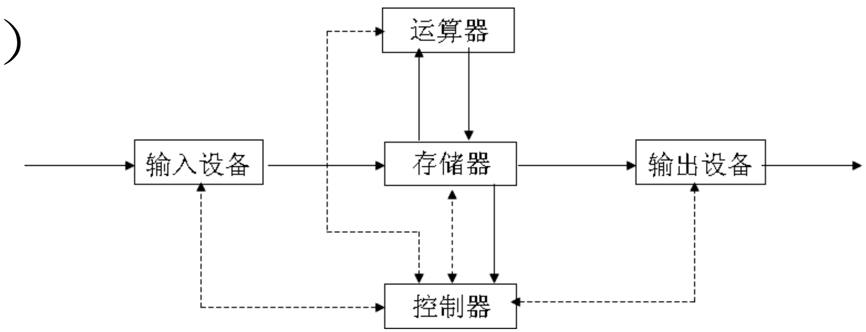
- 以运算器为中心
- 以存储器为中心
  - 哈佛结构
    - 数据与指令相分离
  - 层次化存储系统
- 互连总线
- 并行

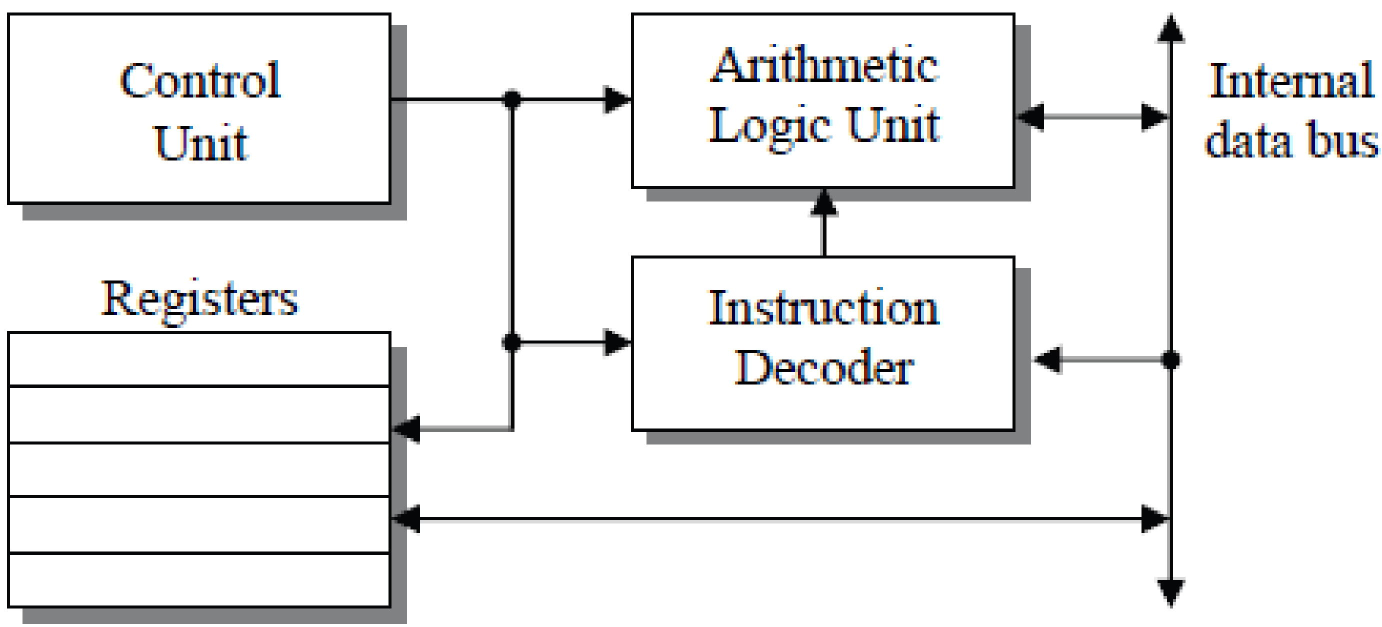




CPU（core）

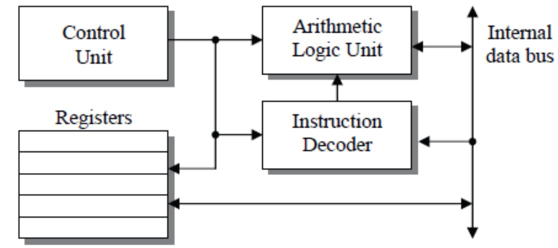
- Generic Block Diagram of a Typical CPU
  - 数据通路datapath（ALU，regs）
  - 控制器：控制信号
  - 行为：数据流？

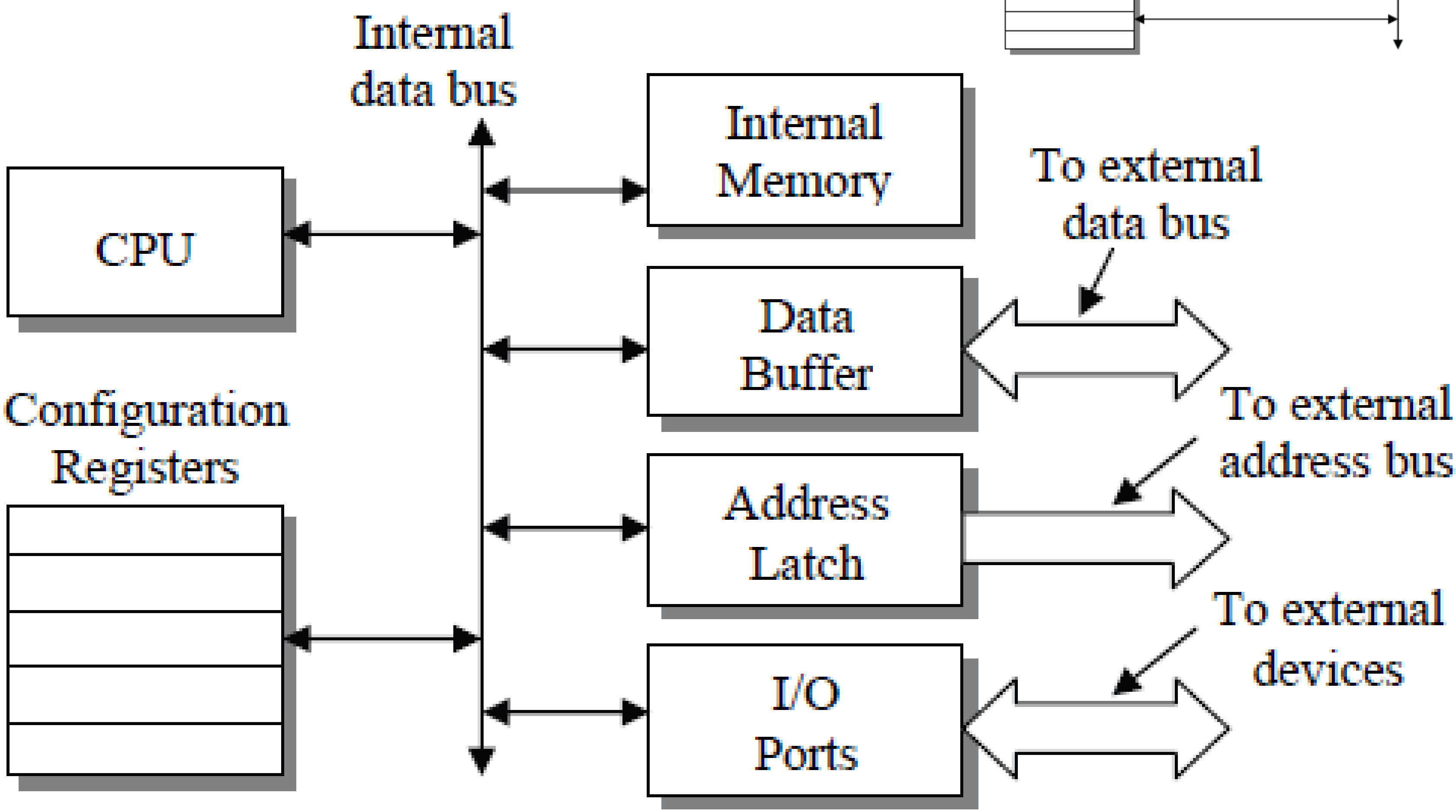




Processor chip【单核，多核】

- Generic Block Diagram





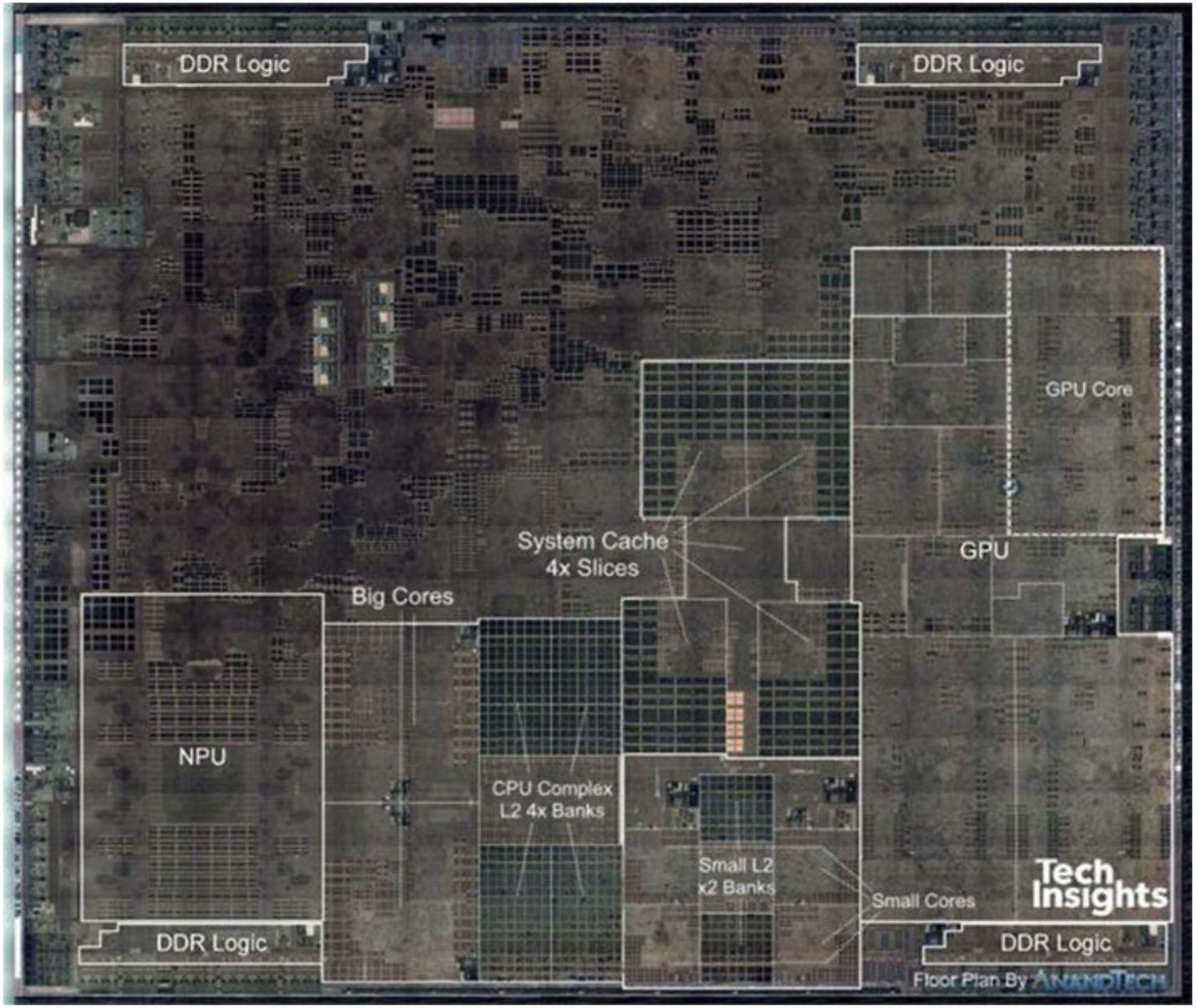
苹果A12（2018发布）版图

8.4 by 9.91 mm
7-nm process
69亿个晶体管
2.49GHz

ARMv8.3-A/64位
bigCore×2
smallCore×4

GPUcore×4
NPU×1

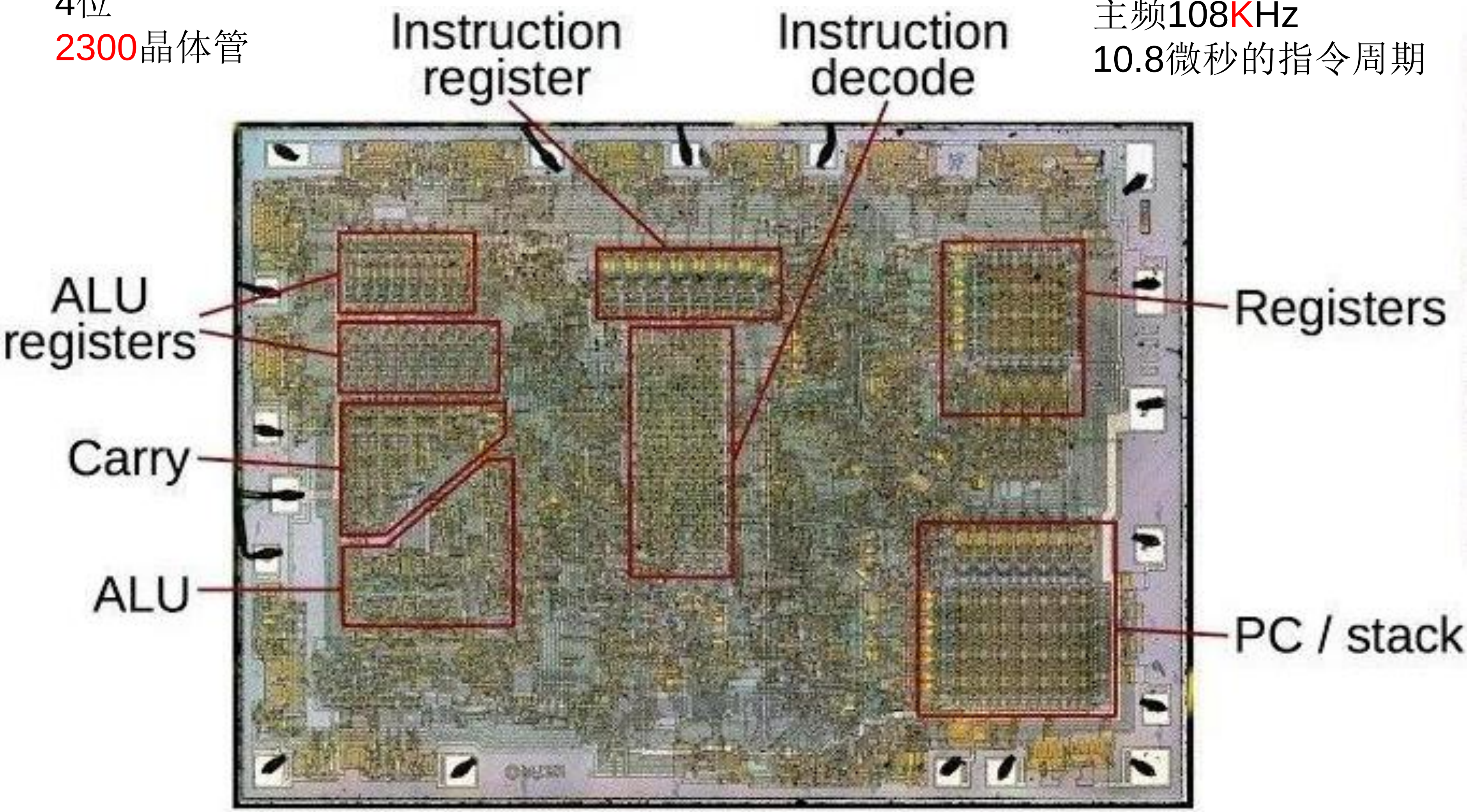
RV32图1-9



Intel 4004 circuit layout，1971

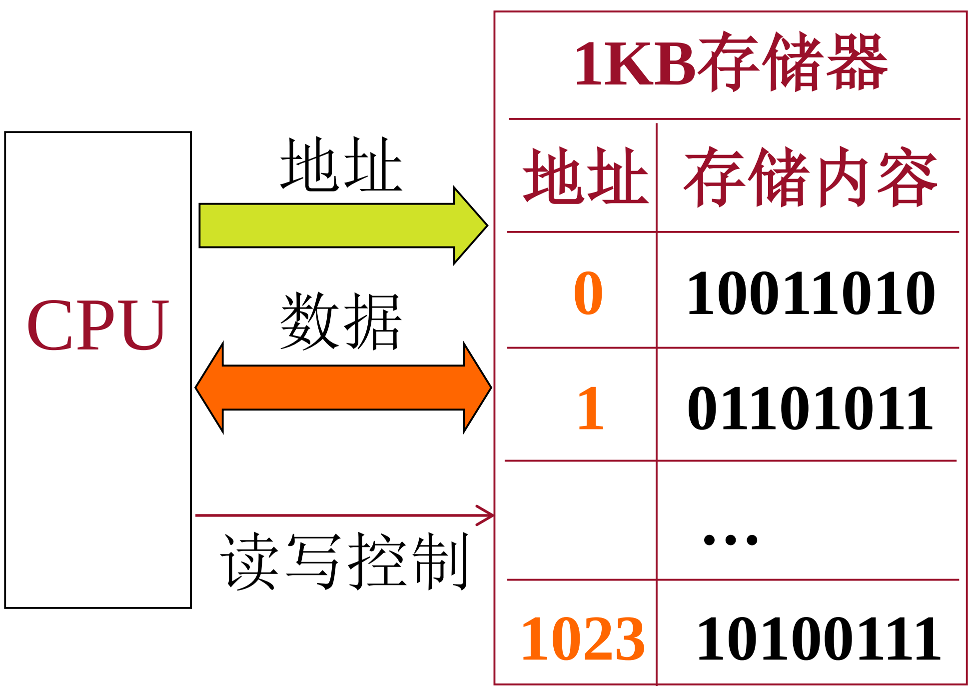
4位
2300晶体管

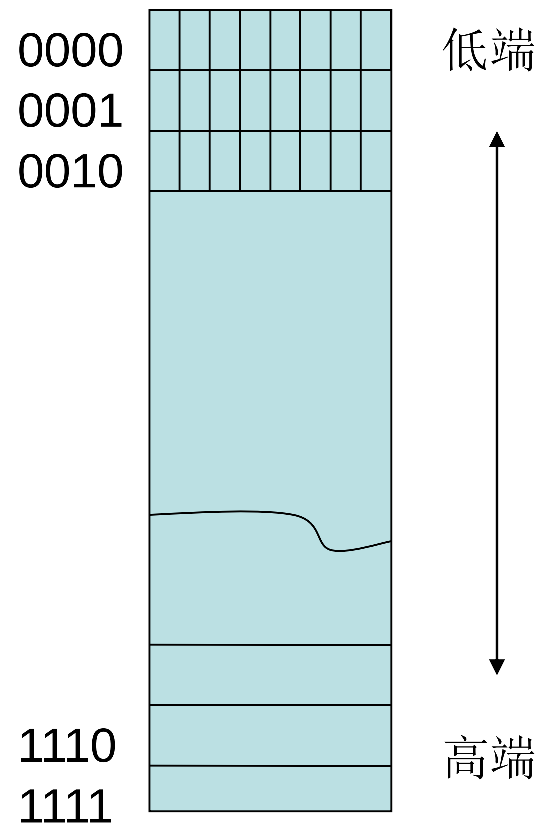
主频108KHz
10.8微秒的指令周期



存储器的组织

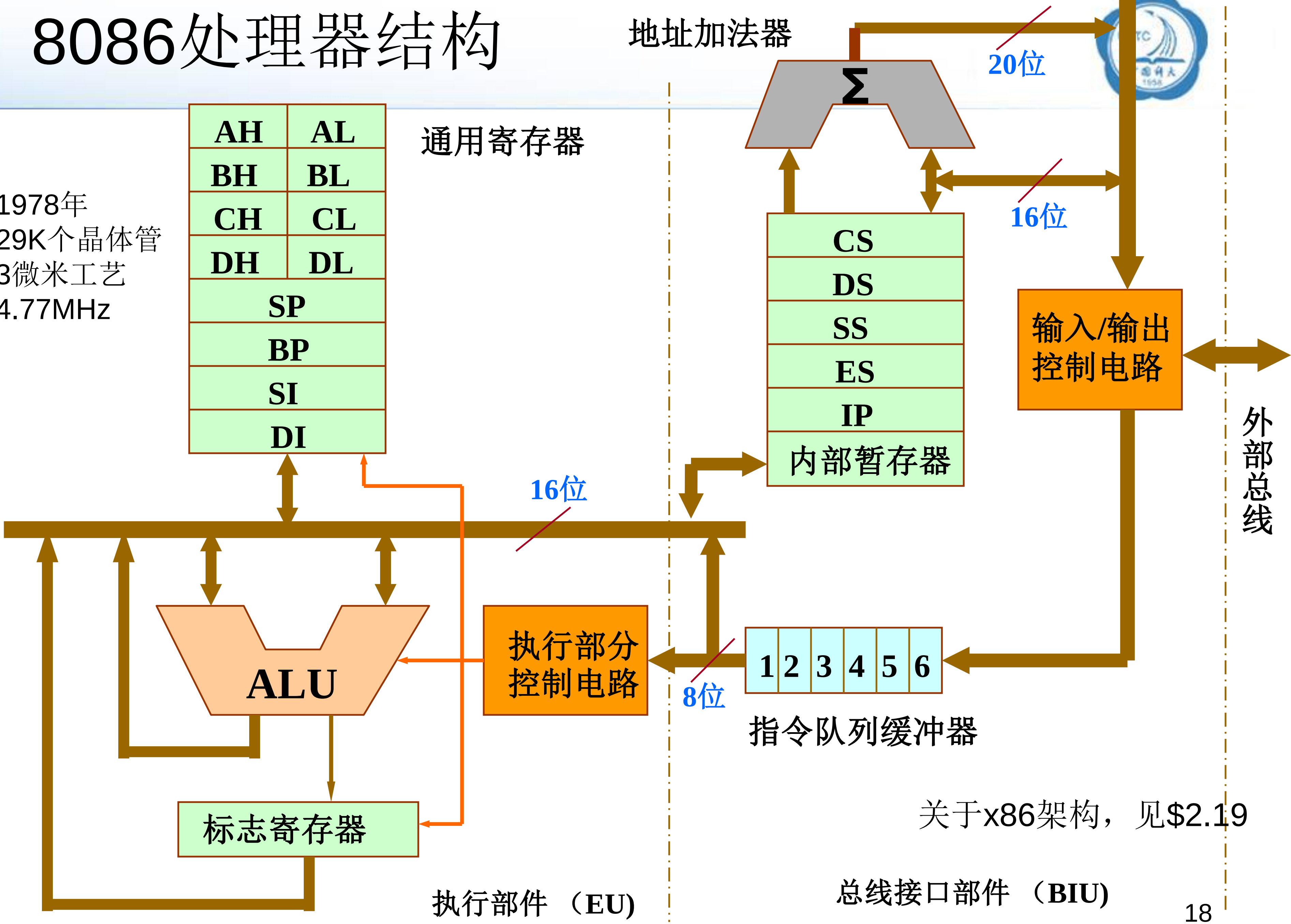
- 存储单元按字节或字寻址【位？】
- 程序和数据顺序存放
  - 数据段：DS寄存器
  - 代码段：CS寄存器
- 读写操作以**数据总线**宽度为单位





8086处理器结构

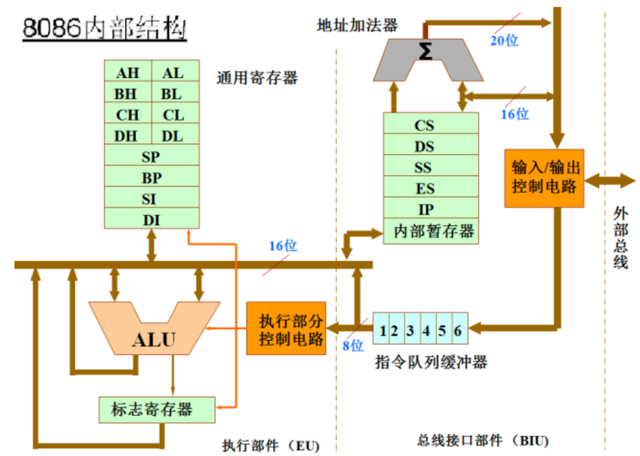
1978年
29K个晶体管
3微米工艺
4.77MHz



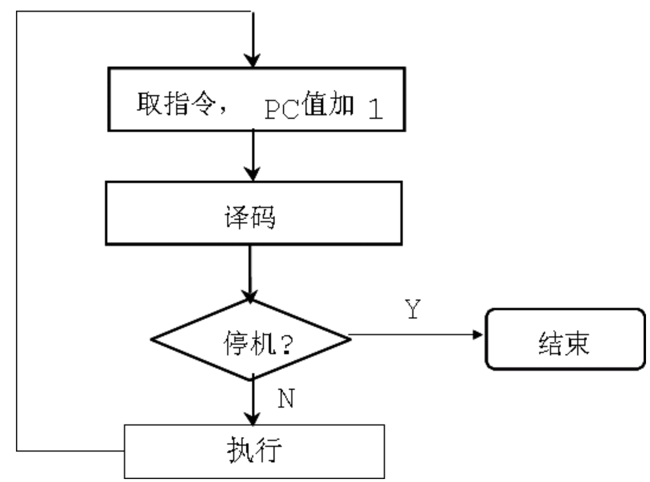
关于x86架构，见$2.19

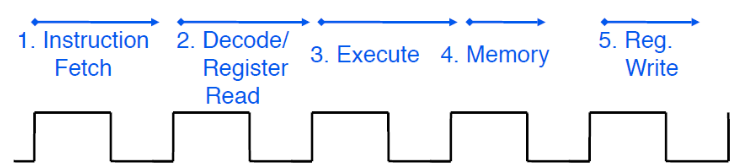
CPU功能与指令执行过程（ABC）

- 指令译码与执行：顺序存储，顺序执行
  - 算术逻辑运算
  - 数据传输：访存，I/O
  - 异常处理和中断响应
- 定时和控制



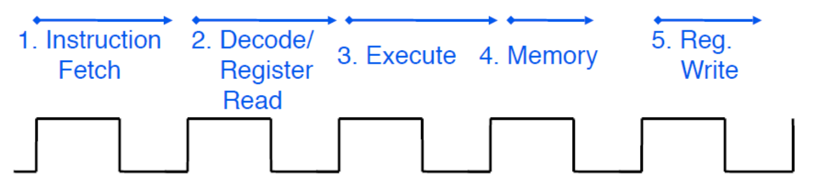
- 取指：IF
  - 根据PC访存读取当前要执行的指令；PC＋1
- 译码：ID
  - 识别指令字中的操作类型，产生控制信号
    - 停机指令？
- 取操作数：OF
  - 根据指令字的地址域读Reg或访存
- 执行：EX
- 写回：WB





程序执行时间：单独执行【！】

- 同步逻辑电路：时钟周期cc（时钟频率），时钟周期数
- 指令的执行时间：指令周期=n个机器周期=m个时钟周期
  - 设：机器周期（访存周期、总线周期）= 时钟周期，即n=m
  - CPI：执行一条指令所需的时钟周期数
    - 平均值：多种指令（计算、分支、访存、I/O、…）
  - IPC
  - 指令执行速率MIPS

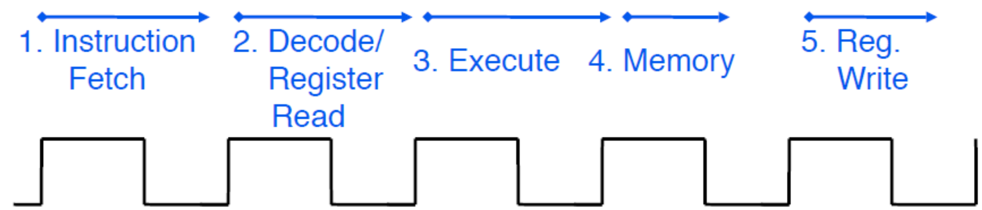


- 程序的CPU执行时间（s）
  - 程序单独运行时所占用的CPU时间=指令数×指令周期宽度
    - CPU时间 = 程序总的CPU时钟周期数 / 时钟频率
    - 程序总的CPU时钟周期数 = 程序的指令数 x CPI
  - 铁律（“Iron Law” of Processor Performance）

  CPU时间 =（程序的指令数 x CPI）/ 时钟频率 = (IC × CPI) / f

时钟周期，机器周期，指令周期

例1：一微处理器，主频为20MHz，请计算其主振时钟周期。若一个机器周期由2个时钟周期组成，平均每条指令用3个机器周期的时间，即“指令周期”，请计算该处理器的平均运行速度MIPS。



[解]

$$时钟周期 = \frac{1}{时钟频率} = \frac{1}{20MHz}$$

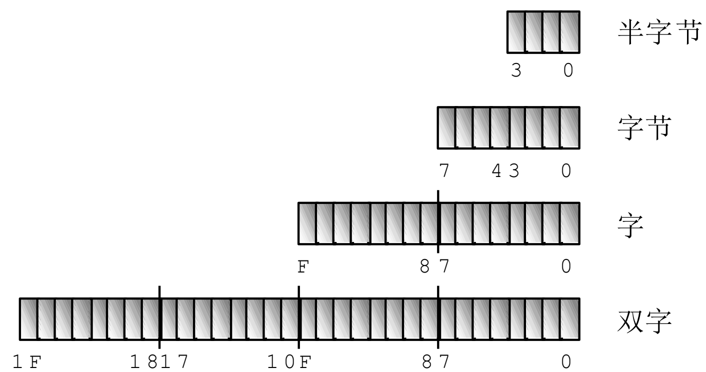
$$= 0.05 \times 10^{-6} S$$

$$平均速度 = \frac{1}{指令执行时间(S)}$$

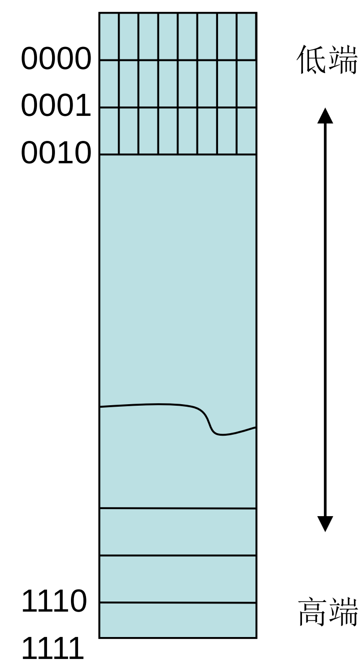
$$= \frac{1}{0.05 \times 10^{-6} \times 6} = 3.33 MIPS$$

计算机**硬件**性能指标（ABC）：$1.6

- 机器字长
  - CPU一次能处理数据的位数
    - 寄存器、ALU、总线、存储器等
  - 字长影响机器数的表示范围和精度
    - 4位、8位、16位、32位、64位



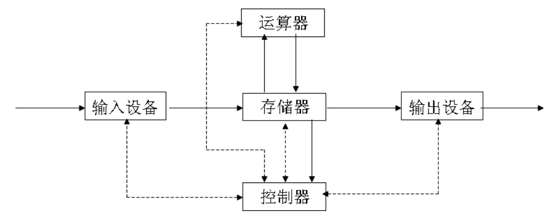
- 内存（主存）容量
  - 存储器可存储的二进制数据总数
    - 容量＝存储单元个数×存储字长
    - MAR(内存地址寄存器)＝16位，则有64K个存储单元；
    - MDR (内存数据寄存器) ＝32位，存储64K×32＝2Mb

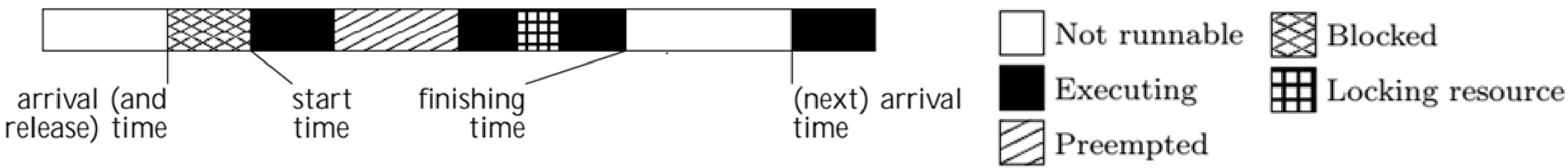


- “运算速度”？
  - 机器应用场景：个人，超算，云计算
  - 用户，设计者
- 功耗

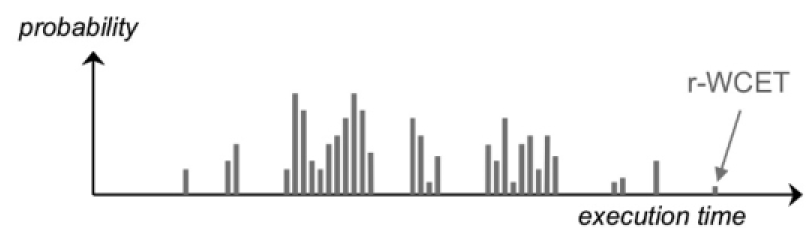
运算速度：执行时间，吞吐率

- 程序执行时间（响应时间，完成时间）
  - 从开始到完成某任务的总时间
  - 含：计算+访存+I/O+OS+资源竞争…

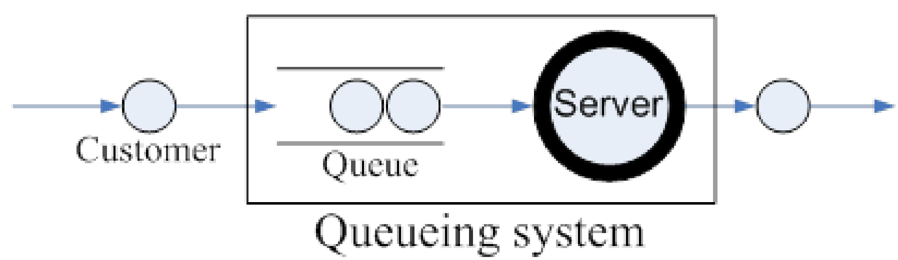




  - 性能 = 1/执行时间
  - 性能比较：A性能/B性能
  - 平均、峰值、最坏



- 吞吐率（带宽）：服务器，网络
  - 单位时间内完成任务的数量
- 两者相互影响
  - “执行时间短，吞吐量高”



度量单位：运算速度、存储容量

- MIPS：每秒百万条指令——指令
- FLOPS：每秒浮点运算次数——操作

图1-1

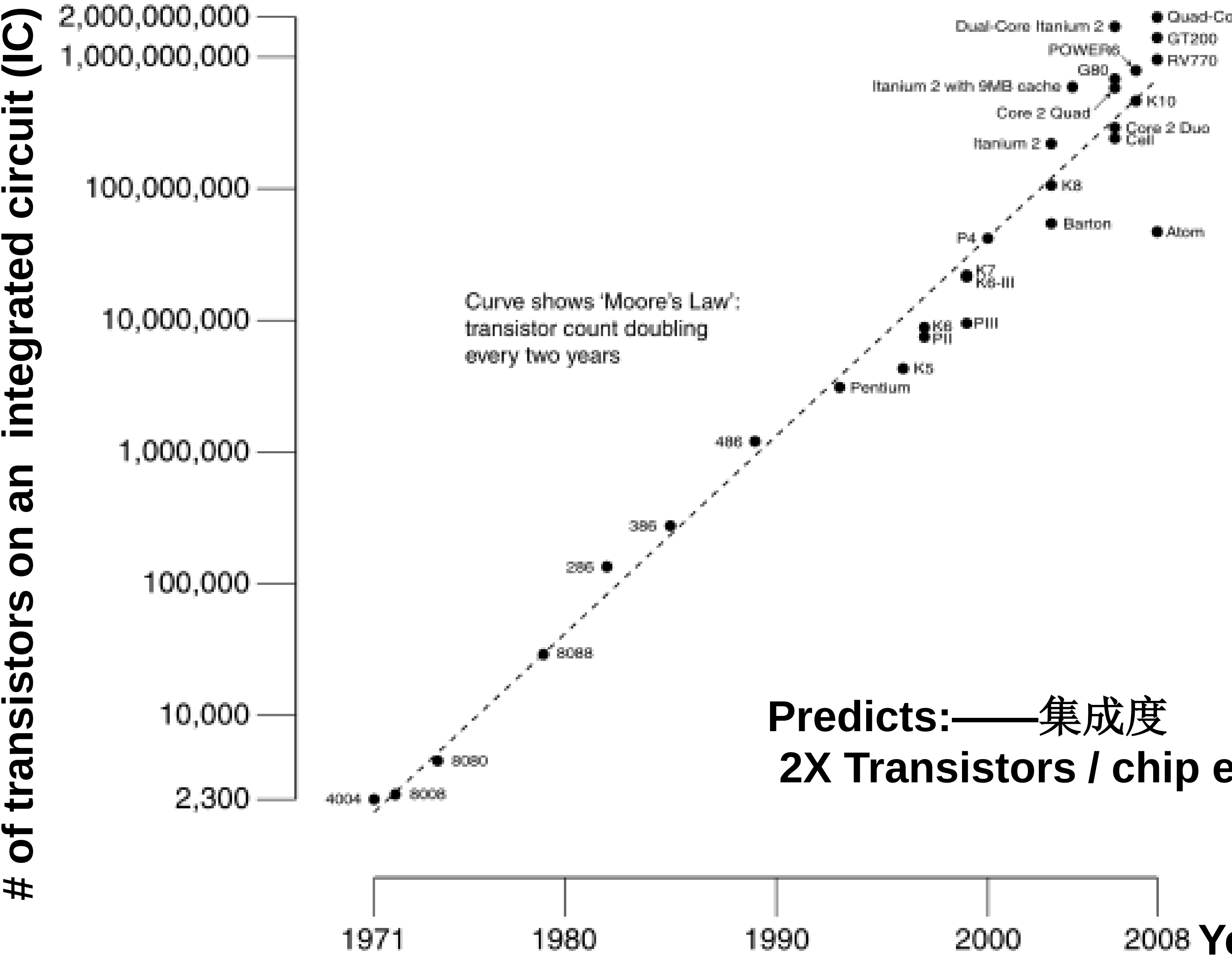
Computer Performance	
Name	FLOPS
yottaFLOPS	10^{24}
zettaFLOPS	10^{21}
exaFLOPS	10^{18}
petaFLOPS	10^{15}
teraFLOPS	10^{12}
gigaFLOPS	10^{9}
megaFLOPS	10^{6}
kiloFLOPS	10^{3}

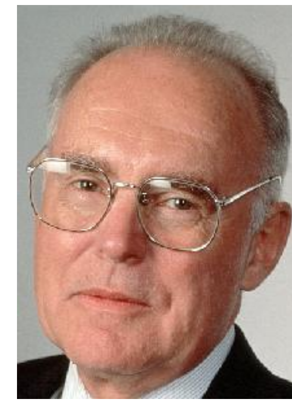
Decimal term	Abbreviation	Value	Binary term	Abbreviation	Value	% Larger
kilobyte	KB	10^3	kibibyte	KiB	2^{10}	2%
megabyte	MB	10^6	mebibyte	MiB	2^{20}	5%
gigabyte	GB	10^9	gibibyte	GiB	2^{30}	7%
terabyte	TB	10^{12}	tebibyte	TiB	2^{40}	10%
petabyte	PB	10^{15}	pebibyte	PiB	2^{50}	13%
exabyte	EB	10^{18}	exbibyte	EiB	2^{60}	15%
zettabyte	ZB	10^{21}	zebibyte	ZiB	2^{70}	18%
yottabyte	YB	10^{24}	yobibyte	YiB	2^{80}	21%

Computility（算力）

- 2021年思元370基于7nm制程工艺，集成390亿晶体管，最大推理算力256TOPS(INT8，Tera Operations Per Second)，训练算力128TFLOPS。【寒武纪】
- 一台超级计算机的算力是1EFLOPS

影响f的因素：Moore’s Law， 1975

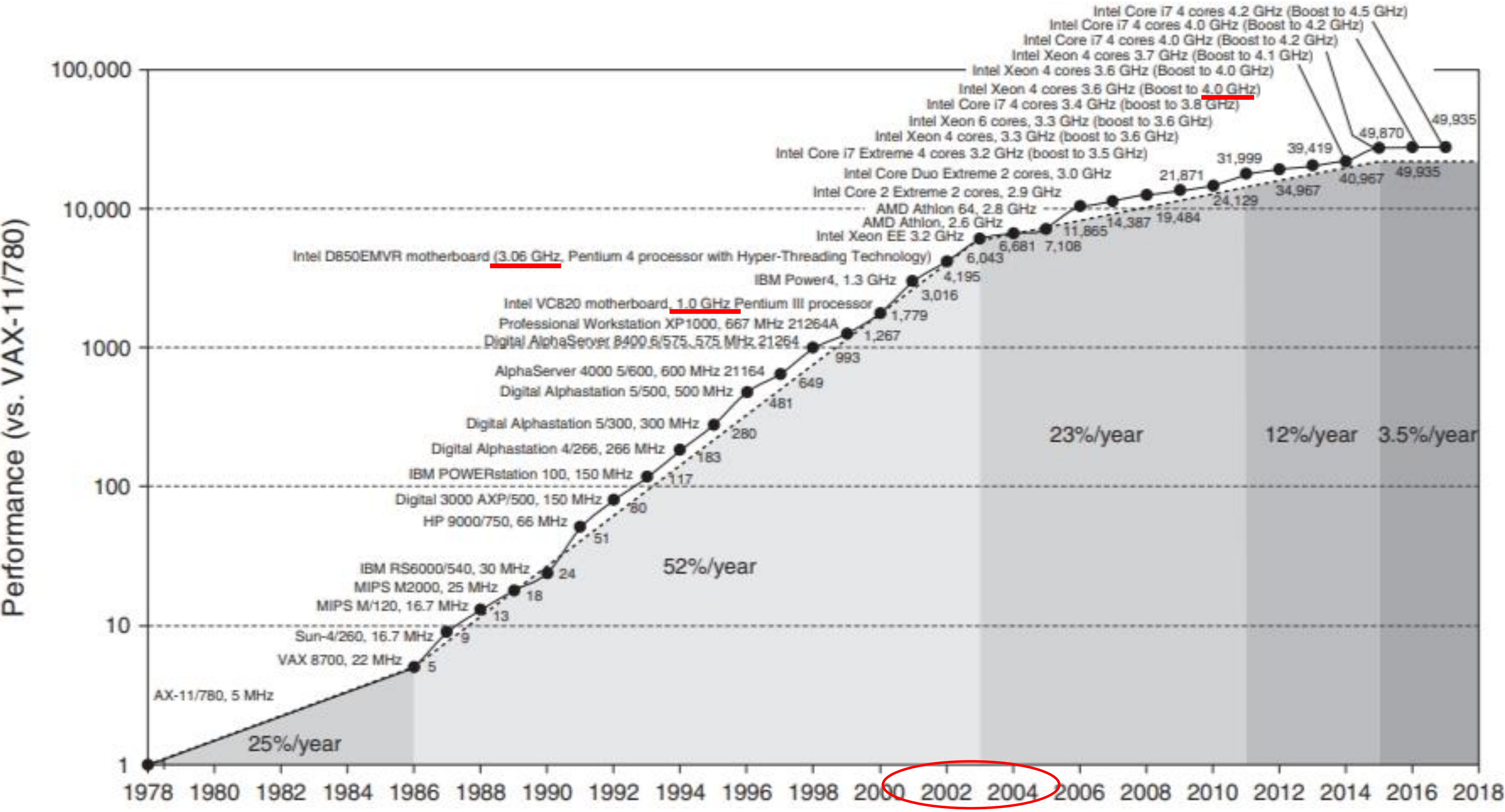




Gordon Moore
Intel Cofounder
B.S. Cal 1950!

Predicts:——集成度
2X Transistors / chip every 2 years

Microprocessor Performance



50% improvement every year!!

What contributes to this improvement?

Source: COD5图1-17

Power Wall: Power Consumption Trends

- dynamic power = α activity x capacitance x voltage^2 x frequency
  – Voltage and frequency are somewhat constant now, while capacitance per transistor is decreasing and number of transistors (activity) is increasing
- Leakage power is also rising (function of #trans and voltage)

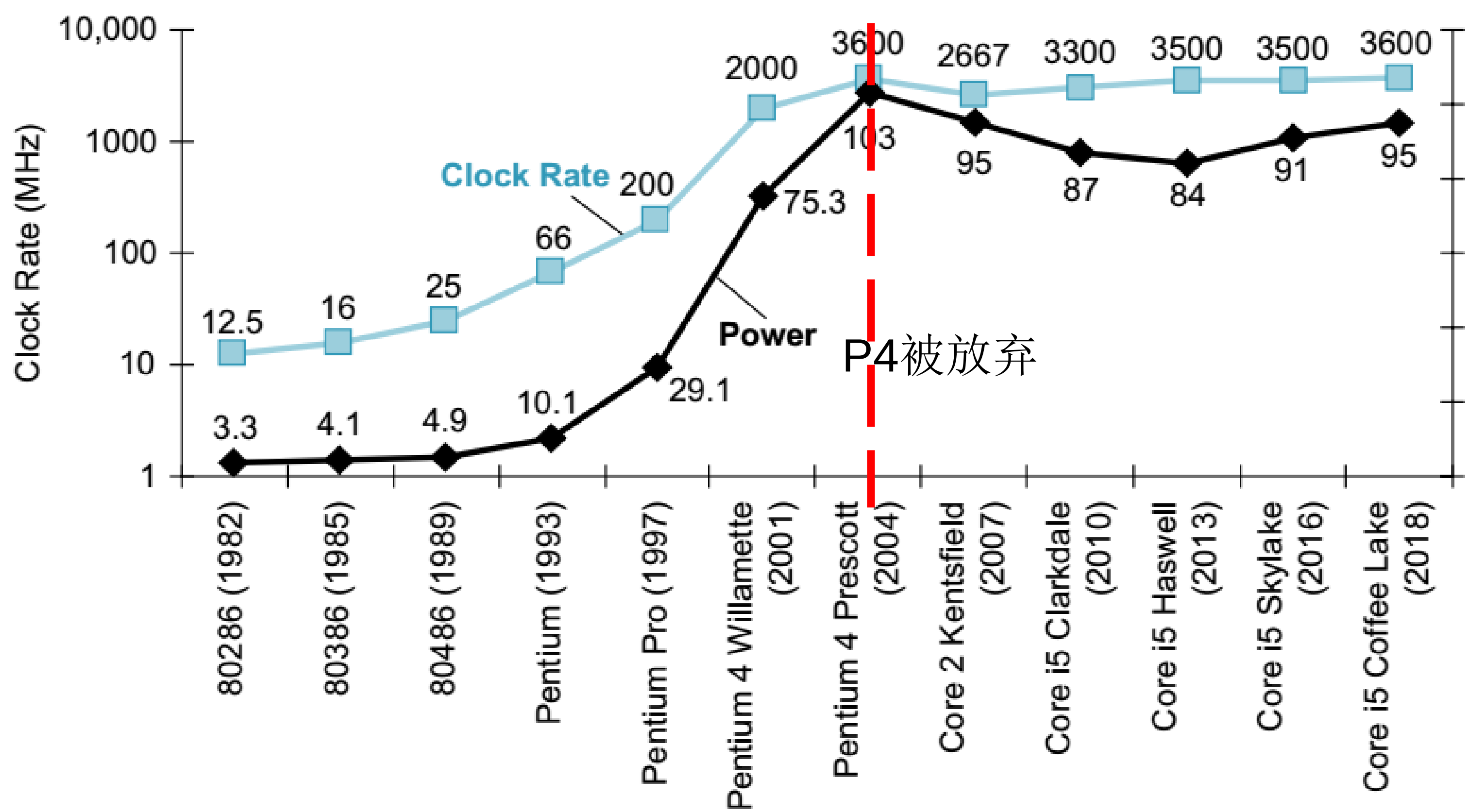
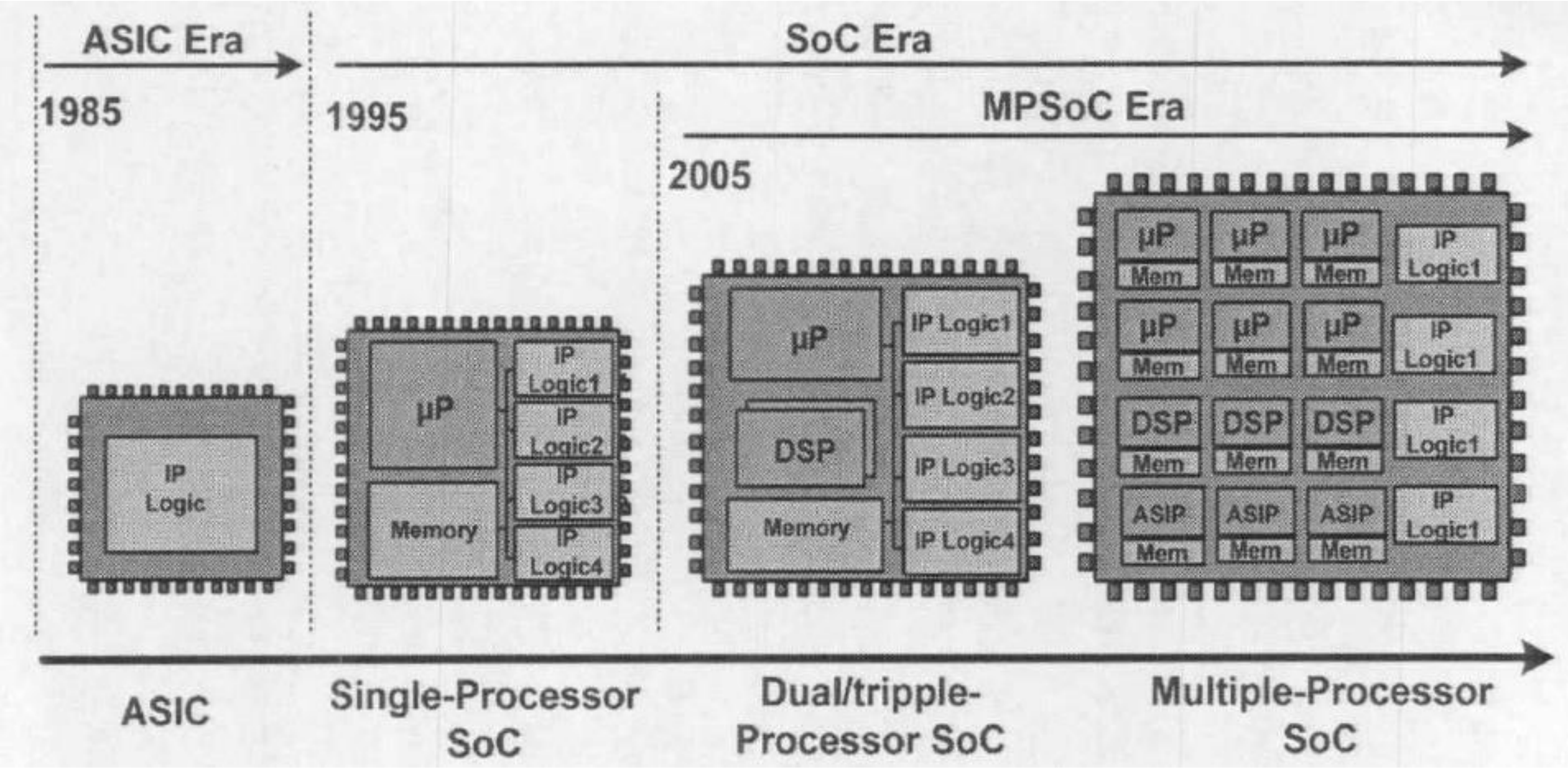


图1-16

多核【SOC Era】



2013，国防科大“天河2”，Top500排名第一，312万个核。1.6万个节点，每个节点195个核（两个intel ivybridge，三个intel phi处理器）

多核/众核有用吗？核数越多越好？生产者-消费者？

并行加速比：Amdahl's Law，1967

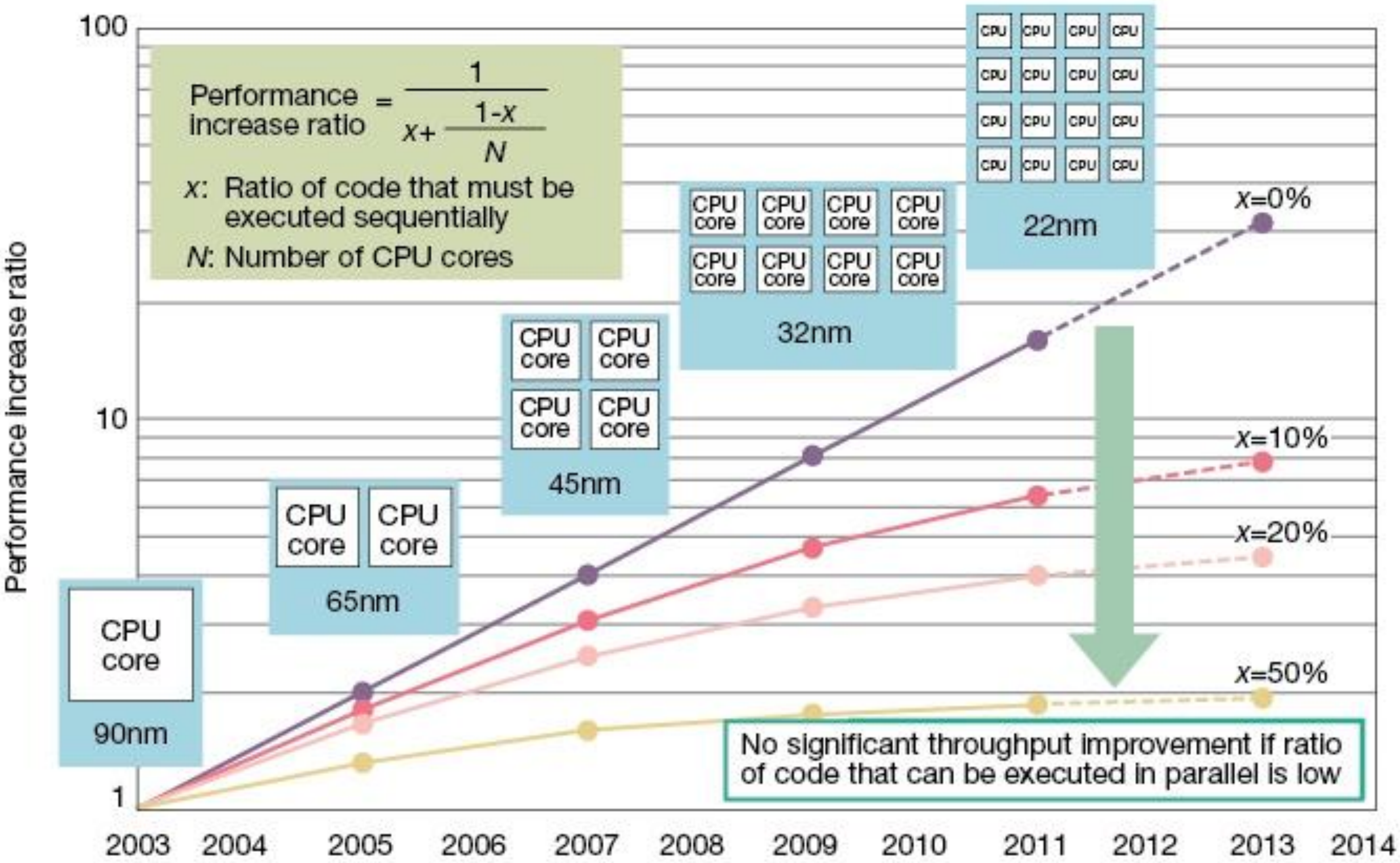


Fig 3 Amdahl's Law an Obstacle to Improved Performance Performance will not rise in the same proportion as the increase in CPU cores. Performance gains are limited by the ratio of software processing that must be executed sequentially. Amdahl's Law is a major obstacle in boosting multicore microprocessor performance. Diagram assumes no overhead in parallel processing. Years shown for design rules based on Intel planned and actual technology. Core count assumed to double for each rule generation.

Gene Amdahl

1）程序执行时间之比
N↑，加速比↑

2）最大加速比
x=0%，加速比=N

3) x影响性能提升！
x↑，加速比↓
x=100%？

4）多核无用？
N=∞， 加速比=1/x

5）缩放性
边际收益递减律

见RV2$1.11、$6.2、$6.13

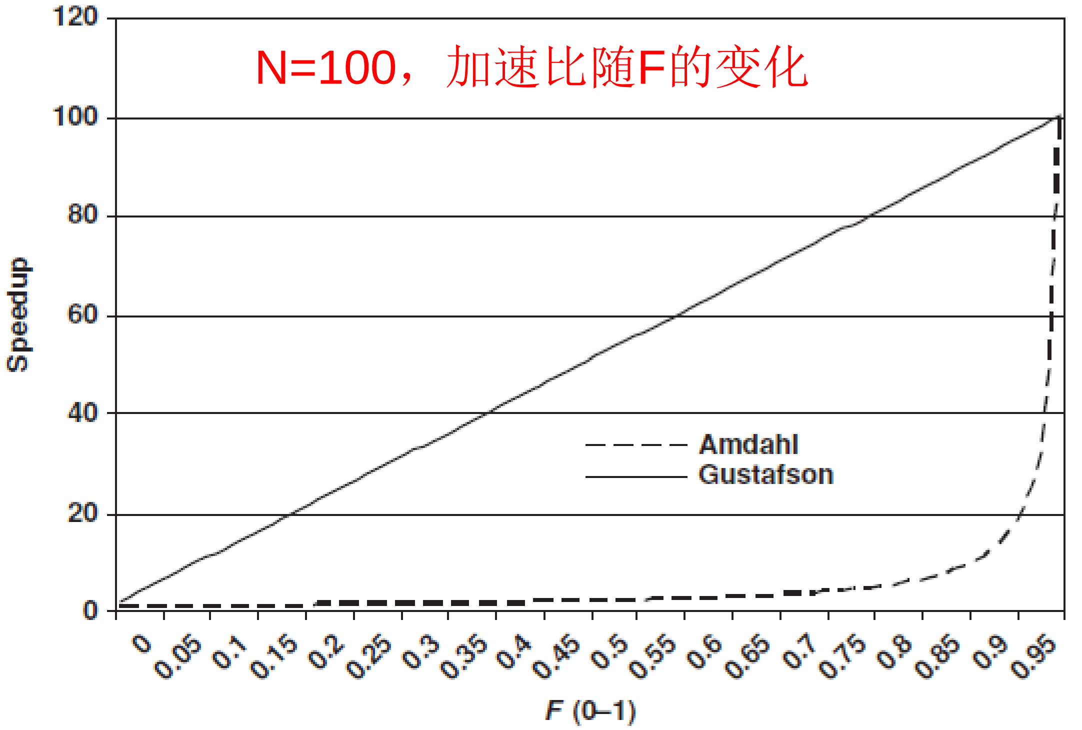
古斯塔夫森定律：1988

- 假设：问题规模与处理器数量同比增长
  - 多核加速比 = 相同时间多核工作量/单核工作量

$$= (s+p*N)/(s+p) = 1 - F + F*N$$

N核数， s=串行时间【≠x】，p=并行时间，F= p/(s+p)=并行时间占比

  - Speedup随着N而线性增长；串行影响性能



John Gustafson

2012 AMD GPU首席架构师

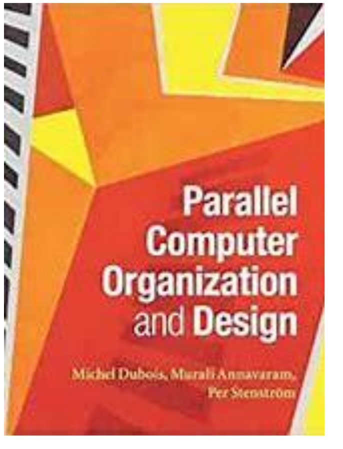


图1-16

Measuring performance（Sys view）

- 执行时间（CPU时间、Elapsed Time）
- 峰值速度（Peak Performance）
- 负载（load）
- 开销 （Overhead）
- 利用率（Utilization Ratio）
- 饱和性能（Saturate Performance）
- 带宽（Bandwidth）
- 延迟（Latency）
- 吞吐率（Throughput）
- 加速比（Speedup）
- 效率（Efficiency）

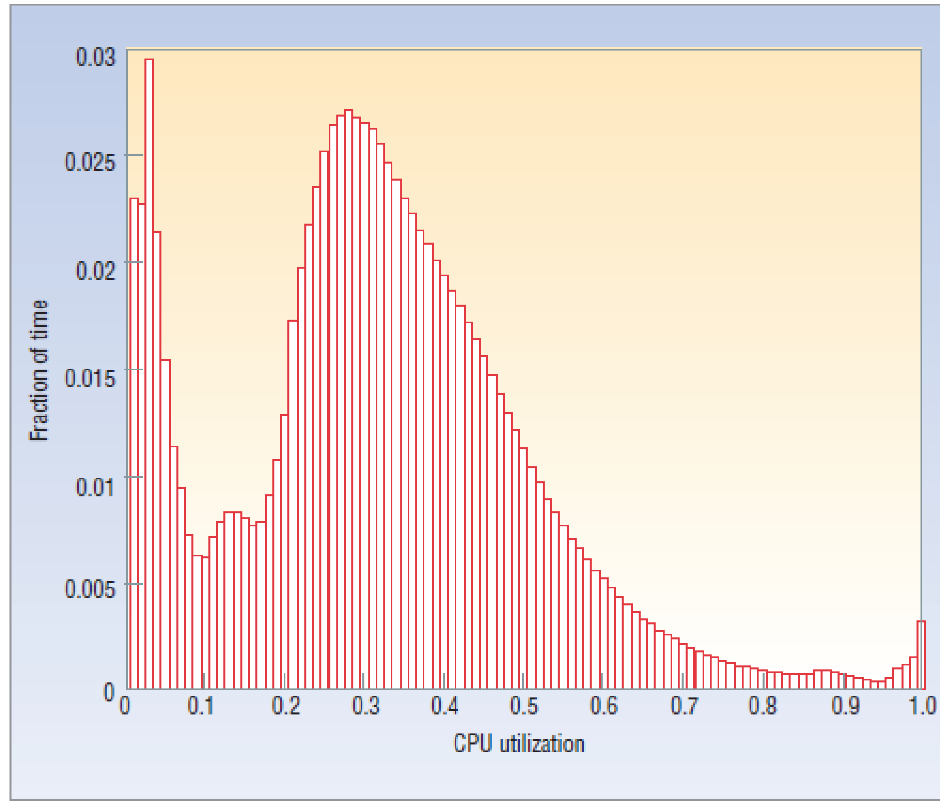
性能测试程序（Benchmark）：$1.9

- SPECcpu2017(System Performance Evaluation Cooperative)
- SPECpower：基于Java基准SPECjbb2005

- Lmbench：操作系统性能，SGI开发
  - 空系统调用时间，进程切换时间，pipe、UDP、TCP、RPC的延迟和带宽，内存、Cache、TLB的读写性能，存储映射的性能
- Webstone：Web服务性能，SGI在1995年开发
  - 吞吐量（MB/s）、延迟（完成一个页面请求的时间）、每分钟传送的页面数、平均连接率、失效率
- Netperf：网络性能
  - 也可用来评测DLPI（Data Link Provider Interface），Unix Domain Socket的性能
  - TCP、UDP的带宽和请求应答数
- SPECsfs97：NFS文件服务器的吞吐量和响应时间
- SPECjvm98：JAVA虚拟机的性能
- SPLASH：共享存储系统性能

PC系统活动与性能分析

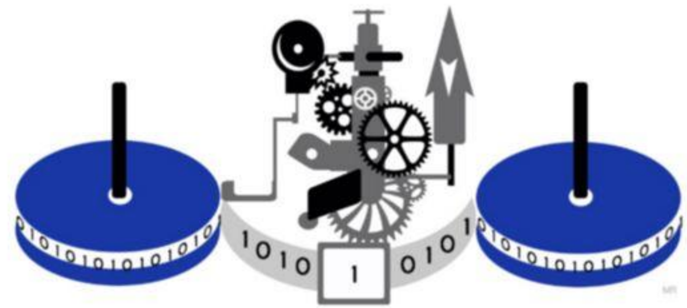
- Window性能分析器
- pcw2008_v186.exe：PC
- CPU_Z
- Intel Battery Life Analyzer
  - 测量CPU利用率
  - 识别高CPU利用率软件部件
  - 测量CPU的C态驻留
  - BLA的USB分析器
- Sysinternals Process Monitor
  - Filemon+Regmon
  - I/O（硬盘、网络）
- Intel VTune



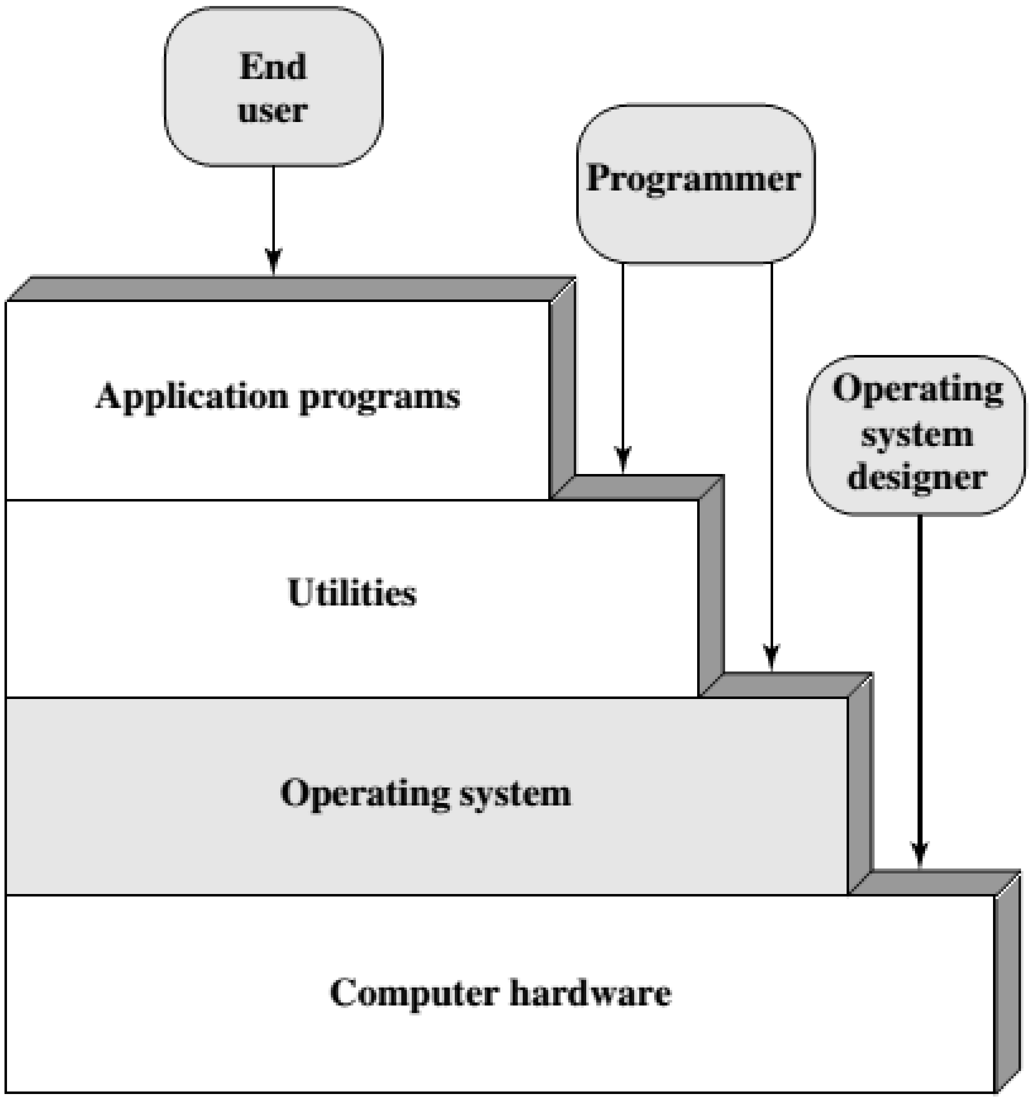
Barroso and Holzle，“The case for energy-proportional computing”. Computer，2007

小结

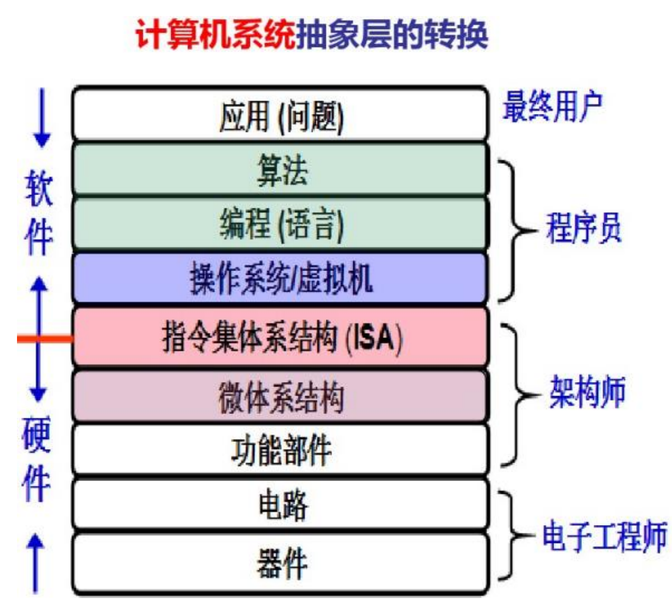
- 计算机系统组成：软件和硬件
- Von Neumann机
  - 计算机硬件系统由五大部件组成
  - 信息以二进制表示
  - 指令由op和addr构成
  - “存储程序控制顺序执行”
- CPU与处理器
  - 数据通路，控制器
- 指令的执行过程
  - 时钟周期，机器周期，指令周期
- 硬件系统技术指标
  - 响应时间，吞吐率
- 加速比



Layers and Views of a Computer System



“计算机在干啥”？



作业

- 作业
  - COD5：1.12，1.13

- 简答题（选做）
  - 图灵机、冯诺依曼机与现代计算机的关系？
  - 冯机架构中指令和数据都存储于存储器中，系统执行时如何区分？
  - “计算机组成”与“计算机体系结构”的关系？
  - 比较Amdahl’s Law和古斯塔夫森定律
  - 主频与计算性能的关系

- 推荐文献
  - Peter Naur, *Turing Lecture*: Computing Versus Human Thinking, *Comm. ACM*, vol. 50, no. 1, pp. 85-94, Jan 2005.
  - *CCC，Arch2030: A Vision of Computer Architecture Research over the Next 15 Years，ISCA2016*
  - John L. Gustafson，"*重新评估阿姆达尔定律，*" 《美国计算机学会通讯》第 31 卷，第 532-533 页，1988 年。

Thank you