# 计 算 机 体 系 结 构

## 第一讲

计算机科学与技术学院

舒燕君

# 主讲教师

- **舒燕君　　容错与移动计算研究中心**
  - 研究方向：
    - ✓ 通用**GPU**性能优化技术
    - ✓ 领域大模型支撑技术
    - ✓ 云边协同计算技术
  - 联系方式：**QQ: 6016511**
  - **Email: yjshu@hit.edu.cn**
  - 办公室地址：综合实验楼**510**房间
  - **https://homepage.hit.edu.cn/shuyanjun**

# 相关课题

- **GPGPU性能优化及领域大模型支撑**
  - ✓ 基于AI编译器的模型推理优化
  - ✓ 面向内存受限的MOE大模型轻量化
  - ✓ 基于AI智能体的大模型微调
- **边缘计算与云计算**
  - ✓ 基于KVM-QEMU的虚拟化实时性研究
  - ✓ 基于Serverless的边缘服务框架
  - ✓ 基于多模态数据云数据中心的AIOps

# 课 程 概 貌

- 讲授内容
  - 现代计算机体系结构基本概念、设计思想、量化分析方法和实现技术

- 教材
  - 王志英等. 计算机体系结构（第2版）. 清华大学出版社，2018
  - 张春元等. 计算机体系结构教程. 清华大学出版社，2025

# 参考书

- **John L. Hennessy, David A. Patterson. Computer Architecture: A Quantitative Approach（第5版），机械工业出版社. 2018**

- **舒燕君等. 计算机组成与结构. 电子工业出版社，2025**

- **胡伟武等. 计算机体系结构. 清华大学出版社，2017**

# 教学模式与课程考核

- 教学模式：采用理论和实践相结合的方法进行教学
  - 40学时课堂教学
  - 24学时实验教学

https://hit-coa.gitlab.io/hit-coa-la32r-lab/cs

- 实验一：设计并实现一个具有五段流水线的**处理器**，并解决数据冲突。
- 实验二：在实验一的基础上，解决控制冲突，并添加一个**动态分支预测器**，提升你的流水线CPU的整体性能。
- 实验三：设计并实现两级流水的**指令Cache**。

# 考核方式与答疑安排

- 课程公告和文件：QQ 群（群号：275492003）

- 答疑安排：
  - ✓ 每周三下午3点-5点，综合楼514。

- 考核安排
  - 期末考试 70%
  - 上机实验 20%
  - 随堂测试 5% 注：一共六次，每次1分，5次则为满分
  - 报告作业 5% 注：一次报告（2分），3次作业（3分）
  - 附加实验 1%-3% 注：EX2 龙芯杯个人赛性能测试

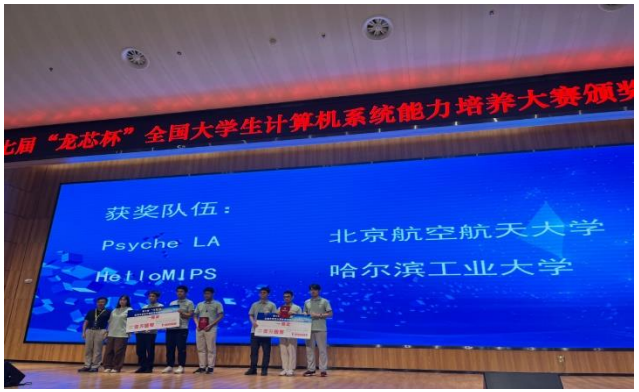
# 助教

- **助教 任婷婷、张诚玮**
  - 负责作业及报告批改、课程答疑
  - 联系方式：课程QQ群

- **助教 王宇杰、徐铎滔、方恒杰、王星哲**
  - 负责指导实验、课程答疑
  - 联系方式：课程QQ群

# 全国大学生计算机系统能力大赛(CPU赛道)

- 团队赛：实现一个LoongArch的CPU，基于龙芯实验平台，下载并进行跑分测试；启动Linux，完成一个较为完整应用设计
- 个人赛：实现一个简单指令集的LoongArch的CPU，基于龙芯实验平台，下载并进行跑分测试。

第 0 章　前言

第 1 章　计算机体系结构基本概念

第 2 章　指令系统

第 3 章　流水线技术

第 4 章　指令级并行

第 5 章　存储层次

第 6 章　输入输出系统

# 跑得最快的计算机（截止2025年6月）

| Rank | Site 国家 | System 名称 | Cores | Rmax (PFlop/s) | Rpeak (PFlop/s) |
|---|---|---|---|---|---|
| 1 | DOE/NNSA/ LLNL | El Capitan - HPE Cray EX255a, AMD 4th Gen EPYC 24C 1.8GHz, AMD Instinct MI300A, Slingshot-11, TOSS<br>制造商：HPE<br>所属：美国能源部Lawrence Livermore国家实验室 | 11,039,616 | 1,742.00 | 2,746.38 |
| 2 | DOE/SC/Oak Ridge National Laboratory | Frontier - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11<br>制造商：HPE<br>所属：美国能源部Oak Ridge国家实验室 | 8,699,904 | 1,194.00 | 1,679.82 |
| 3 | DOE/SC/Argonne National Laboratory | Aurora - HPE Cray EX - Intel Exascale Compute Blade, Xeon CPU Max 9470 52C 2.4GHz, Intel Data Center GPU Max, Slingshot-11<br>制造商：Intel<br>所属：美国能源部Argonne国家实验室 | 9,264,128 | 1,012.00 | 1,980.01 |
| 4 | EuroHPC/ FZJ | JUPITER Booster - BullSequana XH3000, GH Superchip 72C 3GHz, NVIDIA GH200 Superchip, Quad-Rail NVIDIA InfiniBand NDR200, RedHat Enterprise Linux<br>制造商：EVIDEN<br>所属：欧洲高性能计算中心 | 4,801,344 | 793.40 | 930.00 |
| 5 | Microsoft Azure | Eagle - Microsoft NDv5, Xeon Platinum 8480C 48C 2GHz, NVIDIA H100, NVIDIA Infiniband NDR<br>制造商：Microsoft Azure<br>所属：Microsoft Azure | 2,073,600 | 561.20 | 846.84 |

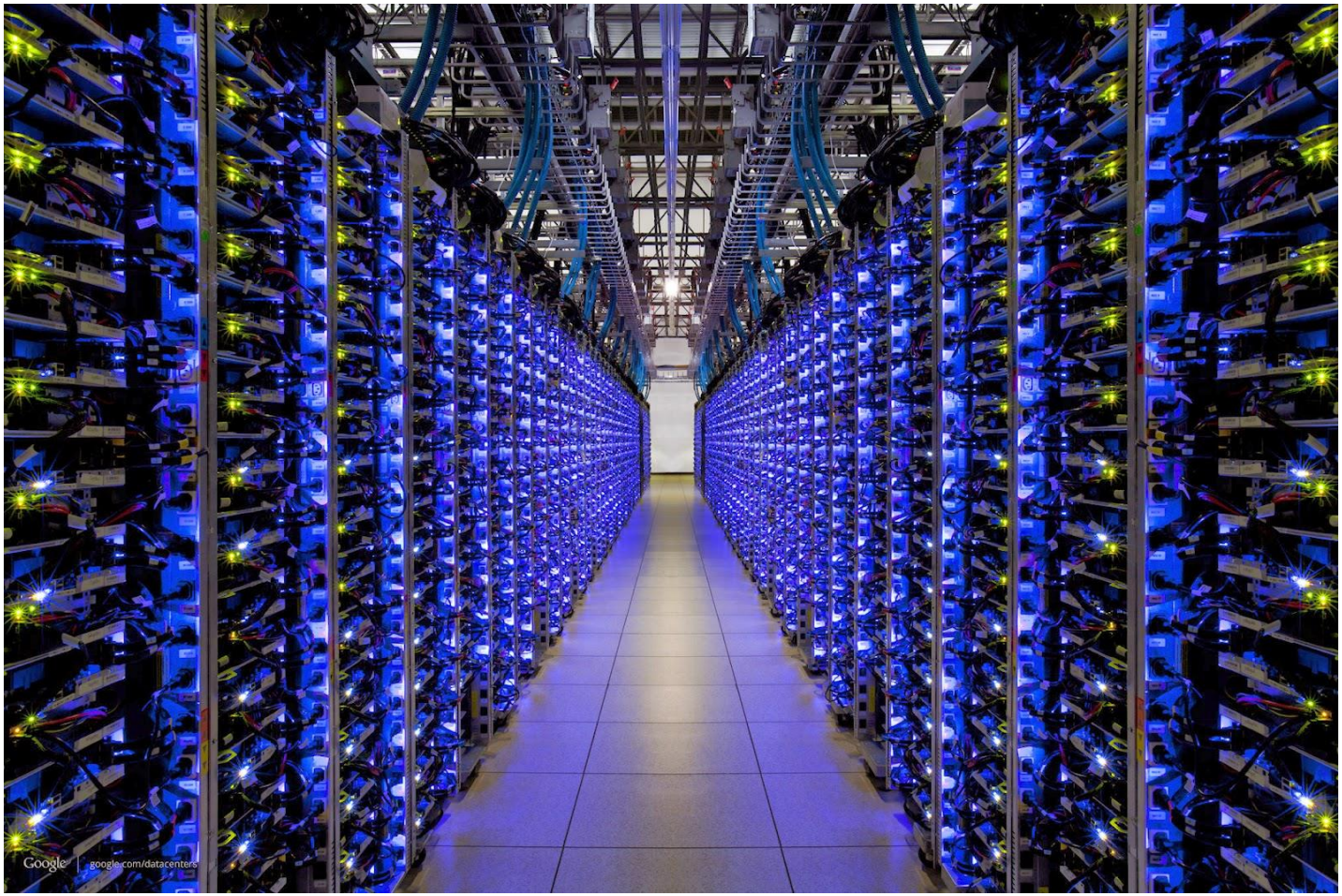
# EL Capitan


| | |
|---|---|
| **Site:** | DOE/NNSA/LLNL |
| **Manufacturer:** | HPE |
| **Cores:** | 11,039,616 |
| **Linpack Performance (Rmax)** | 1,742.00 PFlop/s |
| **Theoretical Peak (Rpeak)** | 2,746.38 PFlop/s |
| **Power:** | 29,580.98 kW |
| **Installation Year:** | 2024 |
| **Interconnect:** | Slingshot-11 |
| **Operating System:** | HPE Cray OS |

# Sunway TaihuLight


| | |
|---|---|
| **Site:** | National Supercomputing Center in Wuxi |
| **Manufacturer:** | NRCPC |
| **Cores:** | 10, 649, 600 |
| **Linpack Performance (Rmax)** | 93,014.6 TFlop/s |
| **Theoretical Peak (Rpeak)** | 125,436 TFlop/s |
| **Power:** | 15,371 kW |
| **Memory:** | 1,310,720 GB |
| **Interconnect:** | Sunway |
| **Operating System:** | Sunway RaiseOS 2.0.5 |

# Different Platforms, Different Goals






Source: http://datacentervoice.com/wp-content/uploads/2015/10/data-center.jpg

# Different Platforms, Different Goals

Source: https://iq.intel.com/5-awesome-uses-for-drone-technology/

# Different Platforms, Different Goals


Source: https://taxistartup.com/wp-content/uploads/2015/03/UK-Self-Driving-Cars.jpg

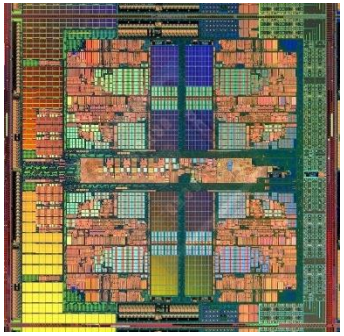
# Different Platforms, Different Goals

Source: http://sm.pcmag.com/pcmag_uk/photo/g/google-self-driving-car-the-guts/google-self-driving-car-the-guts_dwx8.jpg

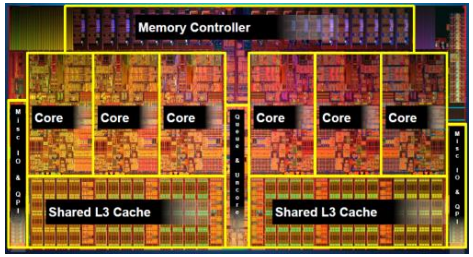
# The Same Applies to Computing Systems

# The Same Applies to Processor Chips

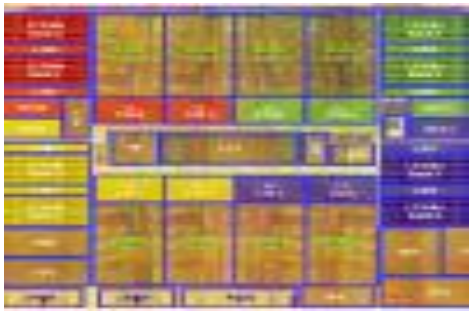
- There are basic building blocks and design principles



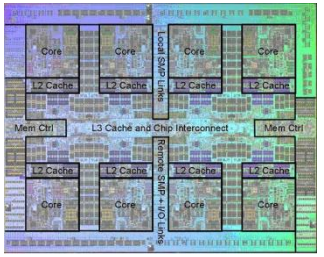
AMD Barcelona
4 cores



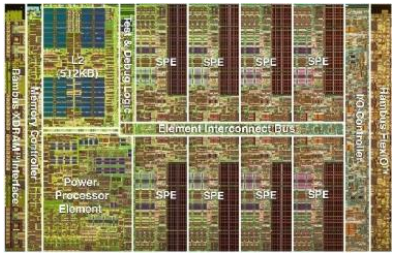

Intel Core i7
8 cores



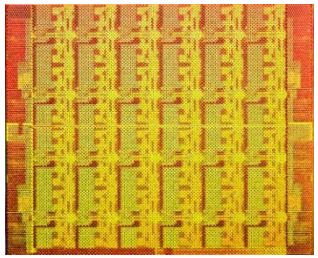
Sun Niagara II
8 cores



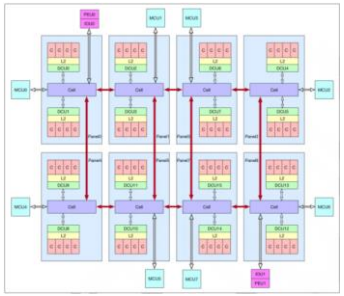

IBM POWER7
8 cores




IBM Cell BE
8+1 cores



Intel SCC
48 cores, networked



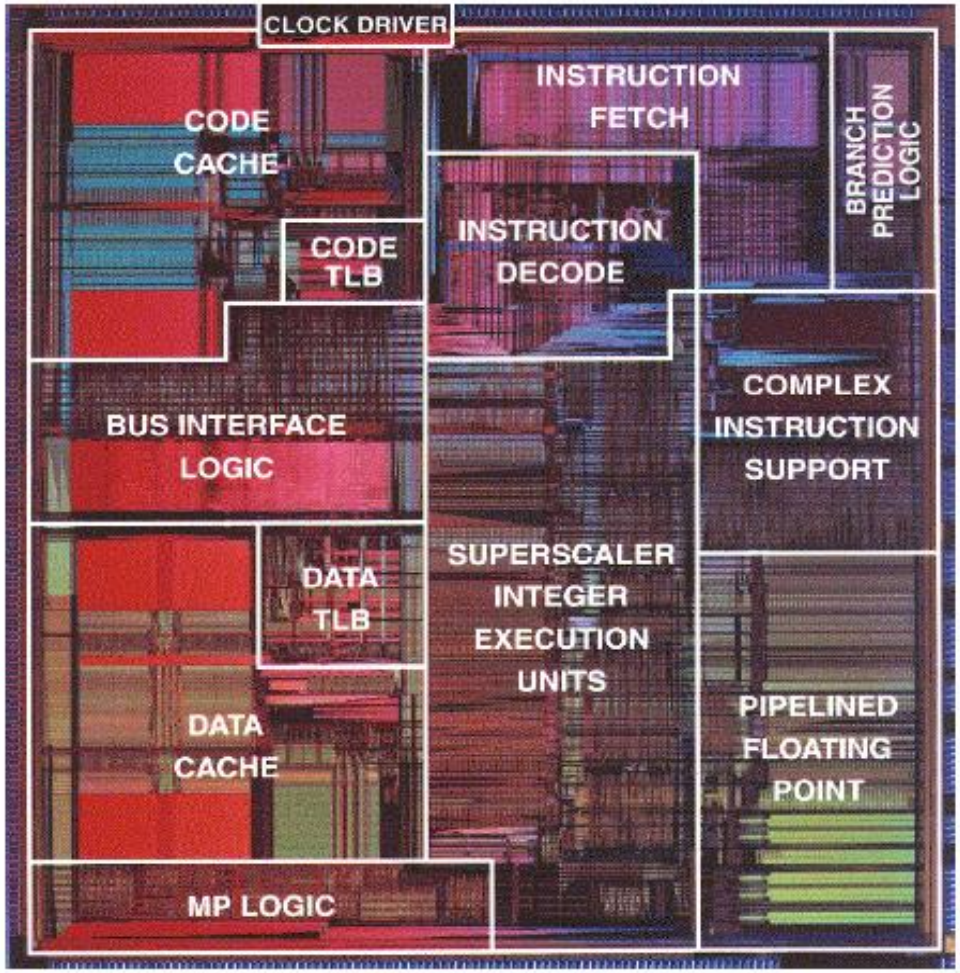
Phytium FT-2000+
64 cores, networked

# Basic Building Blocks

- Electrons
- Transistors
- Logic Gates
- Combinational Logic Circuits
- Sequential Logic Circuits
  - Storage Elements and Memory
- …
- Cores
- Caches
- Interconnect
- Memories
- …

# What Computer Architecture Is All About

- What are the components of a computer?
- How to effectively put together the various components



# 计算机科学

- Computer Architecture
- Computer Software & Theory
- Computer Application

# SYSTEM

Computer Architecture is NOT a hardware course,

BUT SYSTEM!

- Software
- IO System
- Memory system
- CPU

All about how to improve computer system's performance and reduce its cost!

# 为什么要研究系统结构？

# 经典比喻

- 从20世纪40年代末开始，如果交通运输业保持计算机产业的发展速度，今天我们就可以花50美分，在5秒内穿越大洋。（D. A. Patterson & J. L. Hennessy）

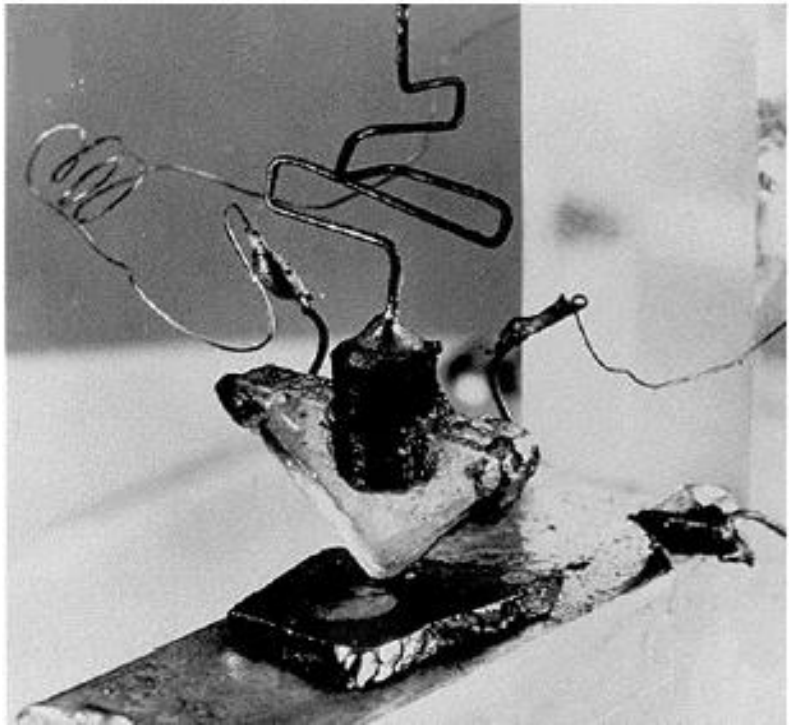
- 如果汽车的性能价格比以同样的速度提升，那么一辆“劳斯莱斯”现在只值1美元，每消耗4.5升汽油，就能行驶16亿公里。（A. S. Tanenbaum, M. van Steen）

# 计算机技术快速进步的原因(1)

- 电子技术进步——集成电路技术的进步，还有存储器（包括内外存）和各类外设的进步。

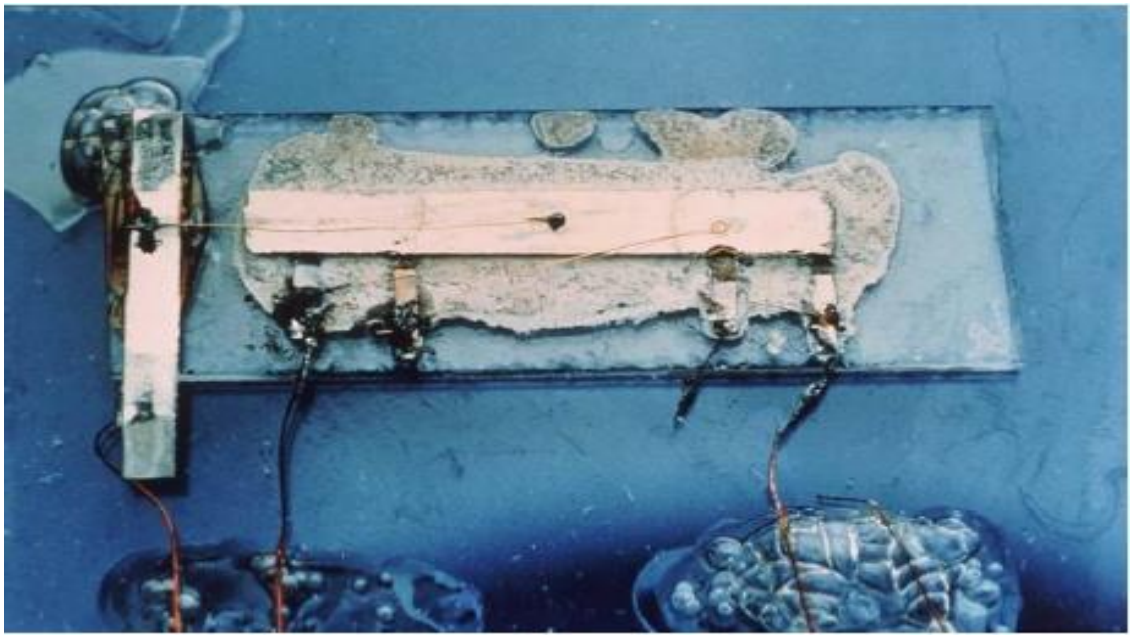
# 电子原件的一大突破—晶体管

- **By W. Shockley, J. Bardeen, W. Brattain of Bell Lab. In 1947**
  - **Much more reliable than vacuum tubes**
  - **Much smaller than vacuum tubes**



# 计算机元件的另一大突破—IC

- **1958年德州儀器公司的Jack Kilby: integrated a transistor with resistors and capacitors on a single semiconductor chip, which is a monolithic IC.**



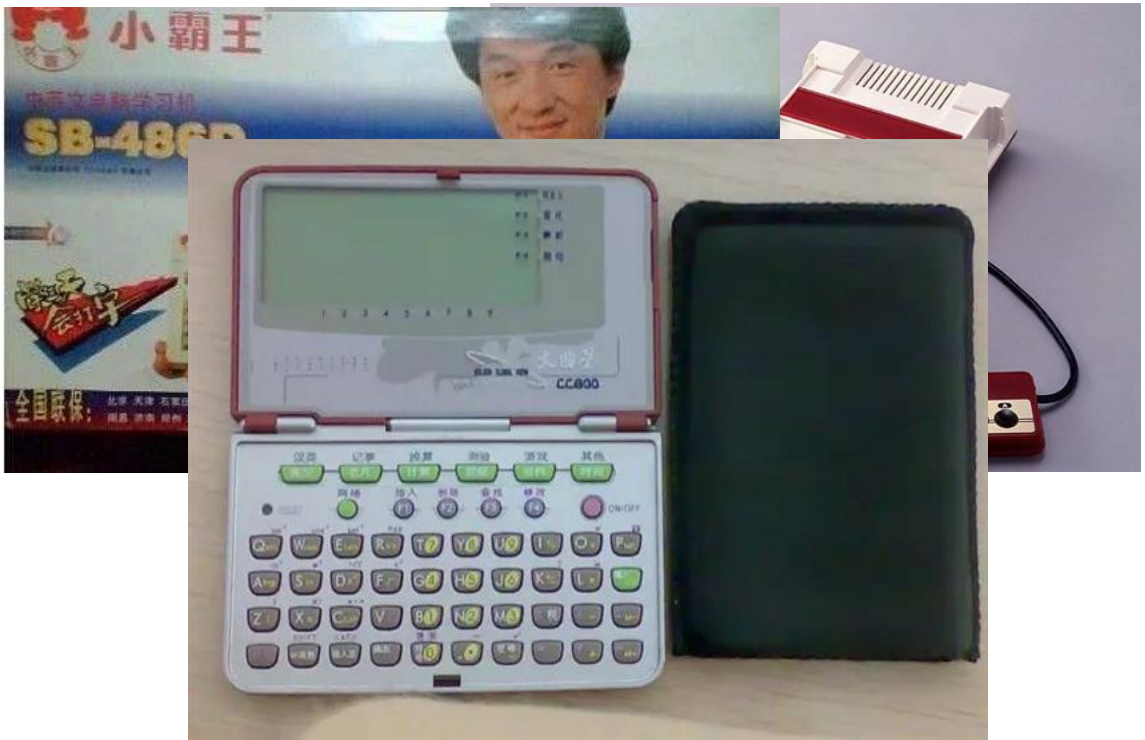
# 计算机技术快速进步的原因(2)

- **计算机技术快速进步的原因二**
  - 计算机系统结构的不断创新
  - 相对制造技术的稳步发展，系统结构的发展总是相对滞后
  - 在电子计算机发展的最初25年中，这两股力量的贡献都很大

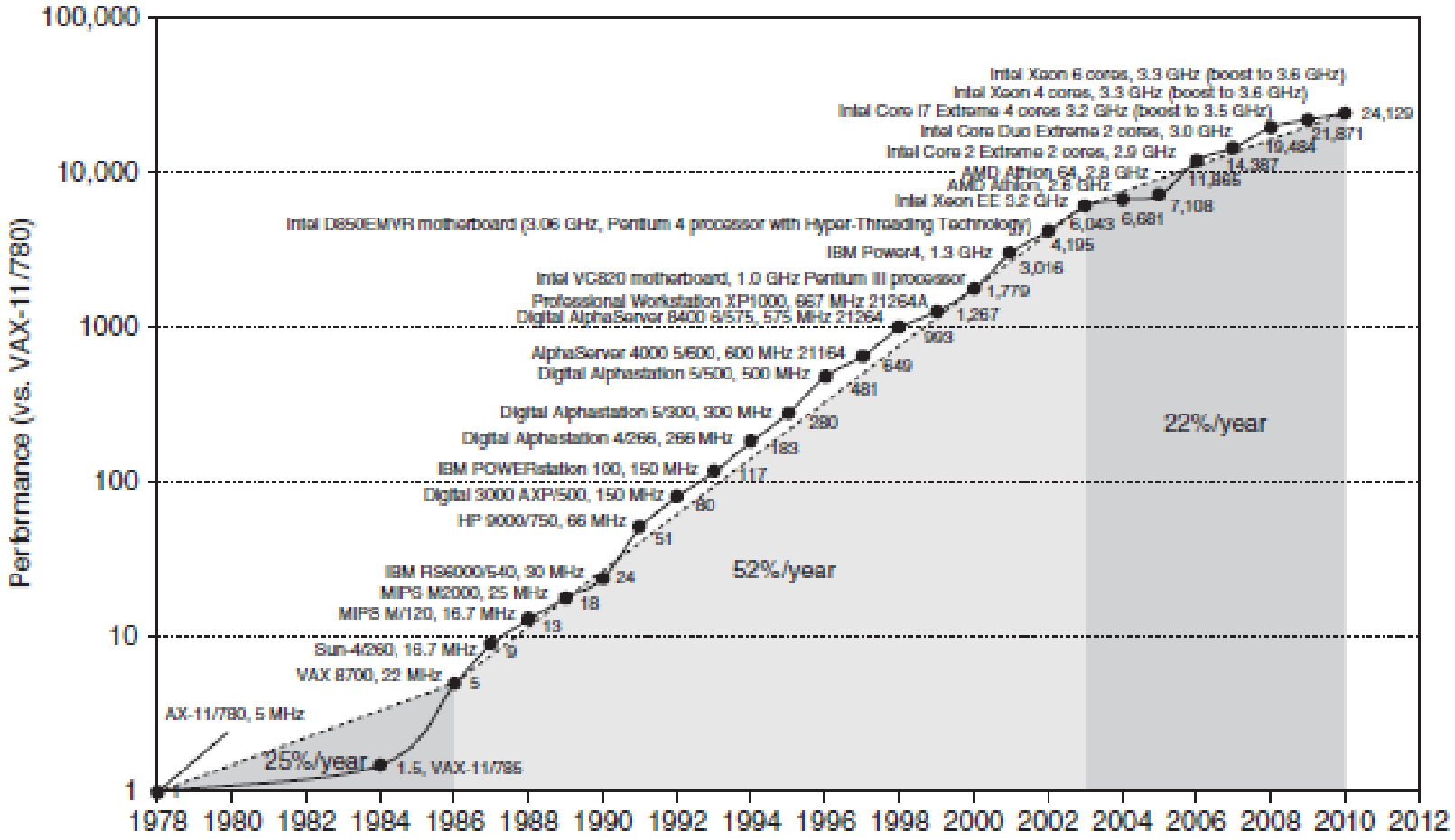
- 大约从七十年代开始，计算机设计者开始更多地依赖于集成电路技术
  - 当时计算机工业占统治地位的大型机和小型机的性能以每年25—30%的速度提高
- 七十年代末出现了微处理器，它比大型机和小型机集成度更高，促进了集成电路技术的发展，计算机性能以大约每年35%的速度提高
- 35%的发展速度，再加上微处理器批量生产的成本优势，使得计算机产业中以微处理器为基础的部分迅速膨胀


- 两个重大的市场变化极大推动了系统结构的发展
  - 人们实际已经极少使用汇编语言编程，这就降低了对目标代码兼容性的要求
  - 标准的、与厂商无关的操作系统（如UNIX）的出现，减小了推出新系统结构的成本和风险
- 计算机市场的两个重大变化使新的计算机系统结构比以往更容易取得商业运作的成功

- 流水
  - 让指令并行起来
    - MOS 6502 取指+执行




- 80年代初，RISC技术出现，80年代中期投放市场，使用RISC处理器的计算机的性能以每年52%的高速度增长
  - 增长速度持续了近16年，截止2002年，微处理器的最高性能与单纯依赖技术进步能够达到的性能相比，前者几乎是后者的7倍




- **长达16年的飞速发展已经告一段落**
  - **2002年起，速度已经降为22%左右**

- 主要因素：
  - 大功耗问题（风冷已到达极限）；
    - As we move from one process to the next, the increase in the number of transistors switching and the frequency with which they switch dominate the decrease in load capacitance and voltage, leading to an overall growth in power consumption and energy
      - The first microprocessors consumed less than 1 watt and the first 32-bit microprocessors (like the Intel 80386) used about 2 watts, while a 3.3 GHz Intel Core i7 consumes 130 watts, Given that this heat must be dissipated from a chip that is about 1.5 cm on a side, we have reached the limit of what can be cooled by air

- 主要因素：
  - 大功耗问题（风冷已到达极限）；
  - 可以进一步有效地开发的指令级并行性已经很少；
  - 难以降低的存储器访问延时（存储器访问速度的提高缓慢）

**系统结构的重大转折：**

从单纯依靠指令级并行转向开发线程级并行和数据级并行。

**Intel从04年开始也取消了单一高性能处理器的研究，通过多核技术进一步提高处理器性能。**

# 为什么要研究系统结构？

半个多世纪以来，计算机技术取得了惊人的发展。1945年时还没有能存储程序的计算机。现在，花不到一千美元买到的个人计算机比1980年花一百万美元买的计算机具有更高的性能、更大的主存和磁盘空间。这一高速发展既得益于计算机制造技术的进步，又离不开计算机设计的创新。

——《计算机系统结构-量化的研究方法》

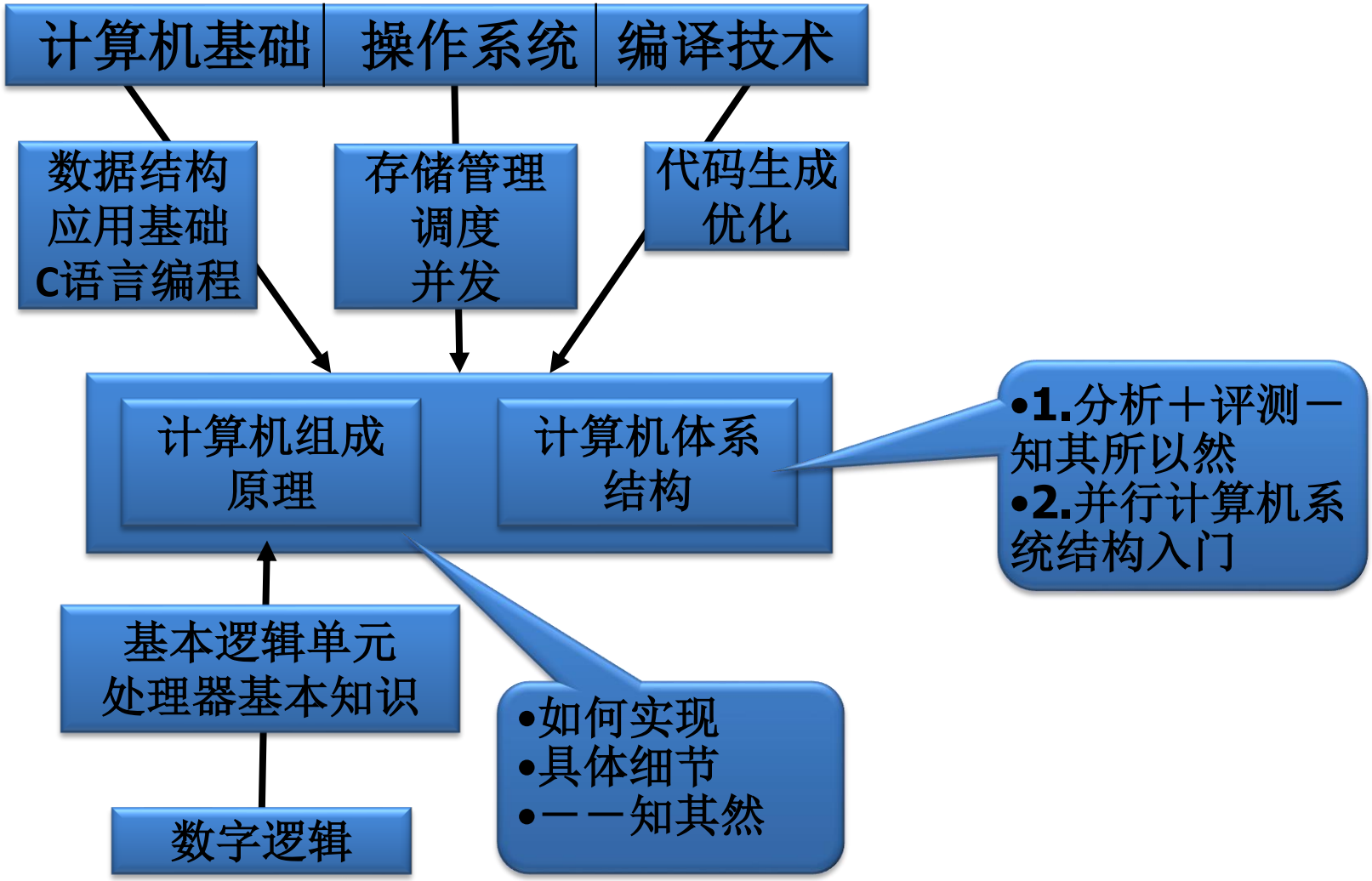
# 学习体系结构的目的是什么？

答案：学习设计计算机的技术！

计算机系统的哪些特征是最重要的，在不超过成本的范围内力求性能最高。

- 建立现代计算机体系结构的基本概念、设计思想和评价技术
- 掌握典型计算机系统的基本结构及其工作原理
- 学习计算机性能分析和评价方法
- 了解先进计算机体系结构设计中的关键技术

# 本课程在课程体系中的地位



# 学习计算机设计技术的必要性

课程主题

每位计算机科学家和工程人员都应该了解计算机的内部机理！！

- 为什么：
  - 一些人将设计计算机，制造计算机
  - 每个人都将使用计算机
    - 越了解计算机，使用就越有效！
- 有益的收获
  - 如何使程序运行的更快
  - 应用程序需要哪种硬件的支持
  - 技术、结构如何变化

# 学习计算机设计技术的必要性（续）

- 放弃微处理器的设计与OS的研究和开发曾是我们的国策。
- 现在已反省这一国策，开发自己的微处理器和OS。
- 在今天后PC时代更有必要。因为在后PC时代，计算机的主要作用不再是独立使用的机器，而是一个应用系统或设备的组件（如马达一样），处理器技术是高性能智能设备的核心。

# 本门课的主要内容

- 要学习的内容:
  - 计算机是如何设计的？（a basic foundation）
  - 如何分析计算机系统的性能
  - 影响计算机性能的主要因素 (caches, pipelines, GPU)
- 为什么要学本门课?
  - 如果你想称自己为“计算机科学家”
  - 如果你想编写高性能的软件
  - 如果你想提出新的计算机体系结构技术

# 第 1 章 计算机体系结构的基本概念

## 1.1 计算机体系结构的概念

## 1.2 计算机体系结构的发展

## 1.3 计算机系统设计和分析

# 1.1 计算机体系结构的概念

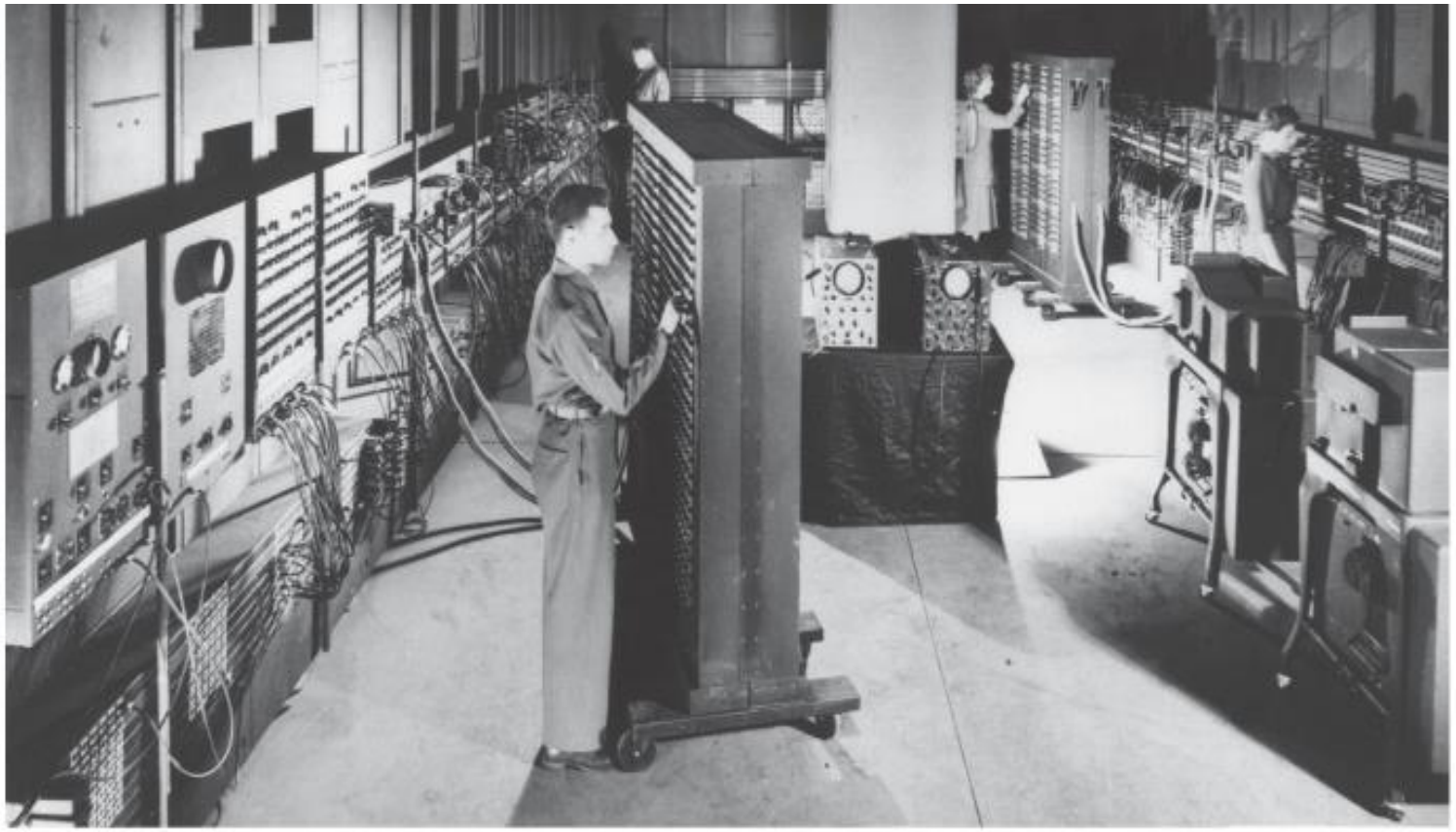
## 1.1.1 计算机体系结构概念的演变

## 1.1.2 计算机体系结构、组成和实现

## 1.1.3 系列机和兼容

# 世界上第一台电子数字计算机ENIAC

- 1945年诞生于美国宾夕法尼亚大学的ENIAC，用于计算火炮的弹道



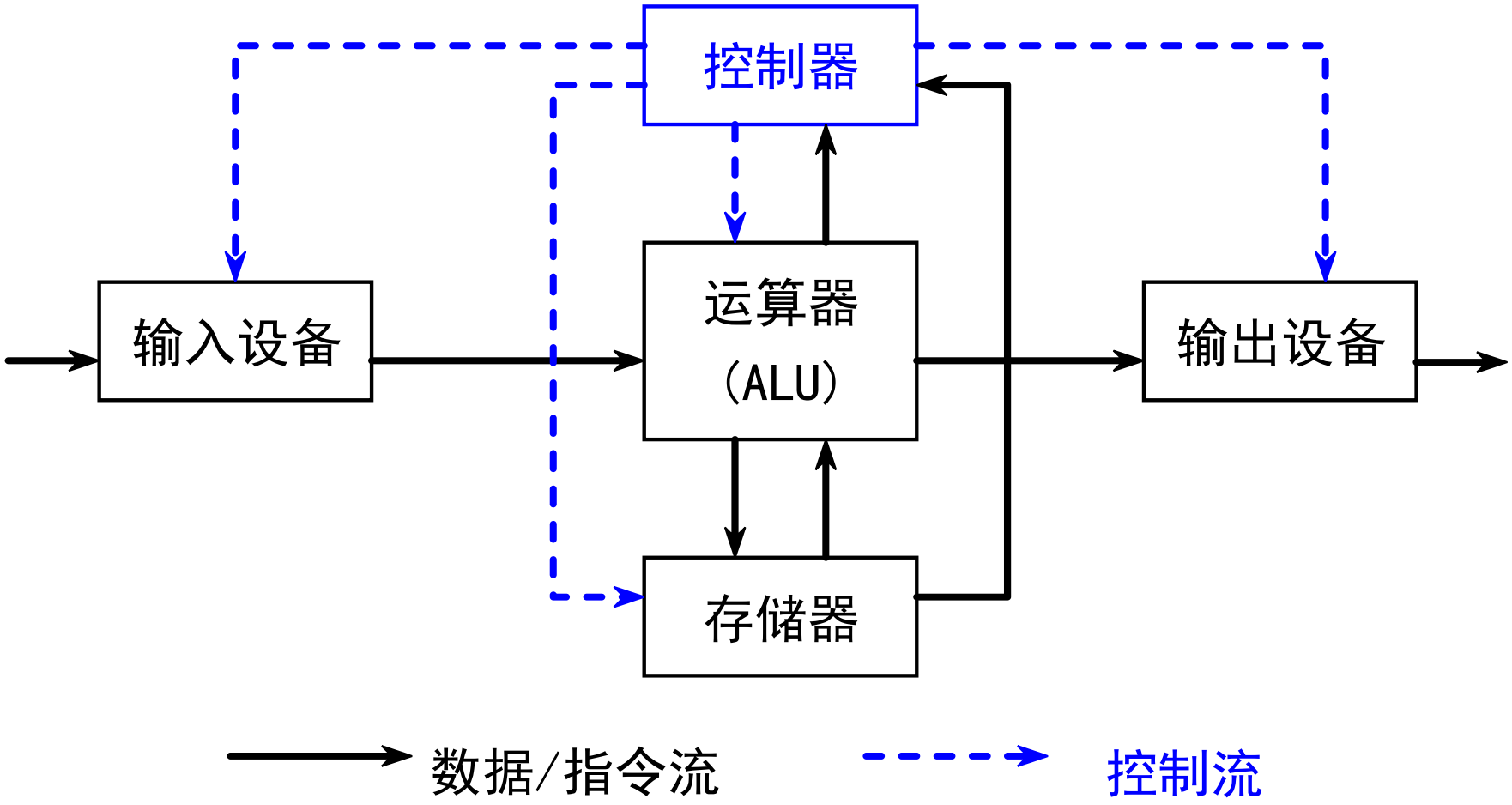
# 冯·诺依曼（John von Neumann）

- 1946年，生于匈牙利的美国数学家冯·诺依曼提出了存储程序计算机
  - Stored-program computer
  - 称存储程序计算机为冯·诺依曼结构计算机
- 计算机的标准结构
  - 70多年来，存储程序计算机的概念和基本结构一直沿用至今，没有发生根本性的变化，是计算机体系结构研究的基础

# 存储程序计算机

- 存储程序计算机
  - 一种计算机系统设计模型
  - 实现了一种通用图灵机
- 冯·诺依曼描述的计算机由五个部分组成
  - 运算器。用于完成数值运算；
  - 存储器。用于存储数据和程序；
  - 输入/输出设备。用于完成计算机和外部的信息交换；
  - 控制器。根据程序形成控制（指令、命令）序列，完成对数据的运算

# 存储程序机器的结构



# 存储程序计算机的主要特点

- 机器以运算器为中心
  - 存储器、输入/输出设备的操作由控制器集中控制
- 采用存储程序原理
  - 程序（指令）和数据放在同一存储器中
- 存储器是按地址访问的、线性编址的空间
- 控制流由指令流产生
  - 解题算法是顺序型的
- 指令由操作码和地址码组成
- 数据以二进制编码表示，采用二进制运算

# 一个机器周期里面安排的操作序列

1. 计算机从存储器中取出一条指令
2. 对这条指令进行译码
   - 分解并确定这条指令所指示的操作
   - 确定操作对象（操作数）所在的位置
     - 某个寄存器单元、存储器单元或者输入设备
3. 取操作数并送到运算器
4. 运算器按照译码确定的操作进行运算
5. 运算结束后，将结果送到指定的位置

- 计算机准备执行下一条指令

# 1.1.1 计算机体系结构概念的演变

- 阿姆道尔（C. M. Amdahl）首次明确
- 计算机体系结构是程序员所看到的计算机的属性，即概念性结构与功能特性
  - 1964年4月，Architecture of the IBM System/360，发表在《IBM Journal of Research and Development》上
  - 计算机体系结构概念的经典定义

# 程序员所看到的计算机的属性

- 对于通用寄存器型机器，这些属性主要是指：

（1）数据表示：硬件能直接辨认和处理的数据类型

（2）寻址规则：最小寻址单元、寻址方式及其表示

（3）寄存器定义：寄存器的定义、数量和使用方式

（4）指令系统：机器指令的操作类型和格式、指令间的排序和控制机构等

（5）中断系统：中断的类型和中断响应硬件的功能等

（6）机器工作状态的定义和切换：如管态和目态等

（7）存储系统：程序员可用的最大存储容量

（8）信息保护：信息保护方式和硬件的支持

（9）I/O结构：I/O寻址方式、数据传送的方式等

# What is Computer Architecture?

- **ISA+Implementation definition:** The science and art of designing, selecting, and interconnecting hardware components and designing the hardware/software interface to create a computing system that meets functional, performance, energy consumption, cost, and other specific goals.
- **Traditional (ISA-only) definition:** "The term *architecture* is used here to describe the attributes of a system as seen by the programmer, i.e., the conceptual structure and functional behavior as distinct from the organization of the dataflow and controls, the logic design, and the physical implementation." *Gene Amdahl*, IBM Journal of R&D, April 1964

# The art of architecture