

计算机组成与结构

人工智能专业

主讲教师：王 娟

《计算机组成与结构》课程综合《计算机组成原理》和《计算机体系结构》两门课程的内容，是“人工智能”专业的核心硬件类课程。

课程目标

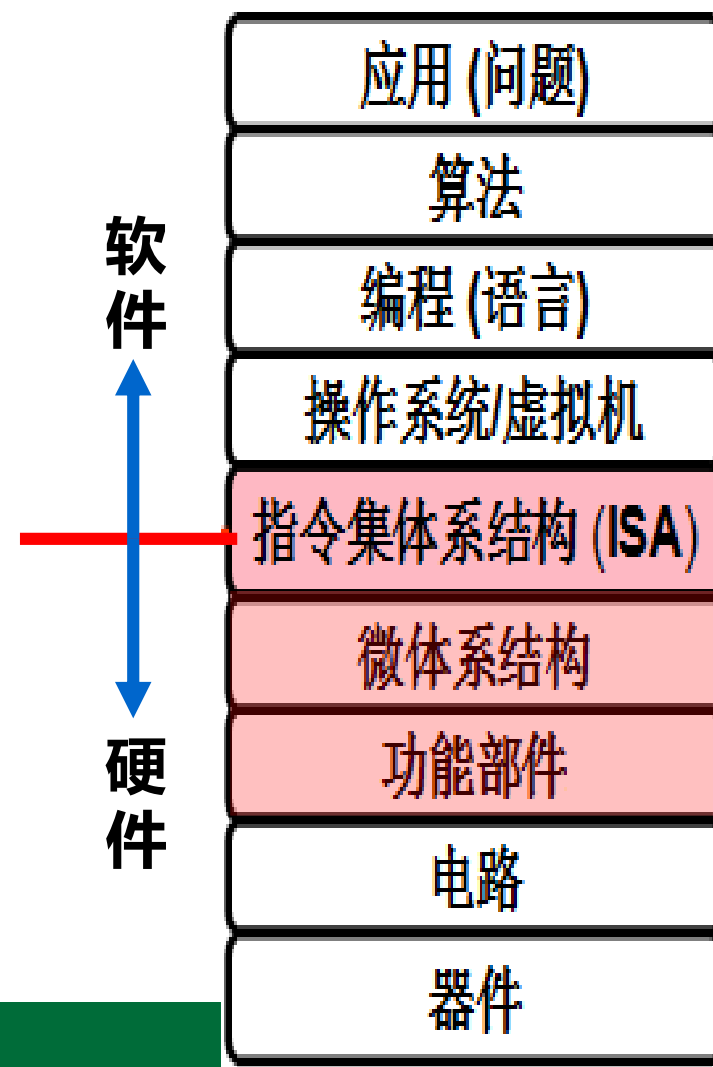
理解**单处理器**计算机系统中各部件的工作原理、组成结构以及相互连接方式，建立完整的**计算机系统**的完整概念。

综合运用计算机组成的基本原理和基本方法，对计算机系统进行分析与设计。

了解计算机系统的最新的发展和研究成果。

计算机系统抽象层

指令集体系结构 (ISA) :
指令系统、机器代码汇编语言
微体系结构:
CPU的通用结构、存储系统
计算机整机系统



课程总学分：2

总学时：32

理论24+实验8

所有课件和作业、实验均在乐学发布。

考核方式：

平时作业、实验和期末考试成绩综合而成。

平时作业成绩：20%，请按时提交，不接受补交。

实验成绩：10%。

期末考试成绩：70%。

【1】蒋本珊，计算机组成原理（第4版），北京，清华大学出版社，2019年。

【2】邝继顺等译，[美]M. Morris Mano, Charles R. Kime 逻辑与计算机设计基础（原书第5版），北京:机械工业出版社，2017年

【3】王党辉等译，[美] David A. Patterson, John L. Hennessy计算机组成与设计-硬件/软件接口（原书第5版），北京：机械工业出版社，2016年9月

【4】龚奕利等译，[美] Randal E. Bryant, david R. O' Hallaron著 深入理解计算机系统（第3版），机械工业出版社，2016 年

【5】贾洪峰等译，[美]John L.Hennessy, David A.Patterson 计算机体系结构：量化研究方法(第5版)，北京：人民邮电出版社，2013年

1. 存储程序与计算机系统组成

2. 计算机的工作过程与计算机性能指标

3. 计算机系统发展历程

存储程序(Stored Program)概念



- (1)计算机（指硬件）应由运算器、存储器、控制器、输入设备和输出设备五大基本部件组成；
- (2)计算机内部采用二进制来表示指令和数据；
- (3)将编好的程序和原始数据事先存入存储器中，然后再启动计算机工作，这就是存储程序的基本含义。

美籍匈牙利数学家冯·诺依曼等人在1945年6月提出存储程序概念。

EDSAC 事实上的第一台存储程序计算机 1949年诞生。

目前绝大多数计算机仍建立在存储程序概念的基础上，称冯·诺依曼型计算机。



世界上第一台电子数字计算机是1946年2月问世的**ENIAC**。

ENIAC的设计开始于1943年, 该机一直使用到1955年。

ENIAC的特点:

- 采用十进制
- 20 个10位的累加器
- 用开关手动编程
- 18,000个电子管
- 重30 吨
- 占地170平方米
- 耗电170 KW
- 5,000次/秒加法运算



计算机系统=硬件系统+软件系统

硬件通常是指一切看得见，摸得到的设备实体；软件通常是泛指各类程序和文件，它们实际上是由一些算法以及其在计算机中的表示所构成的。

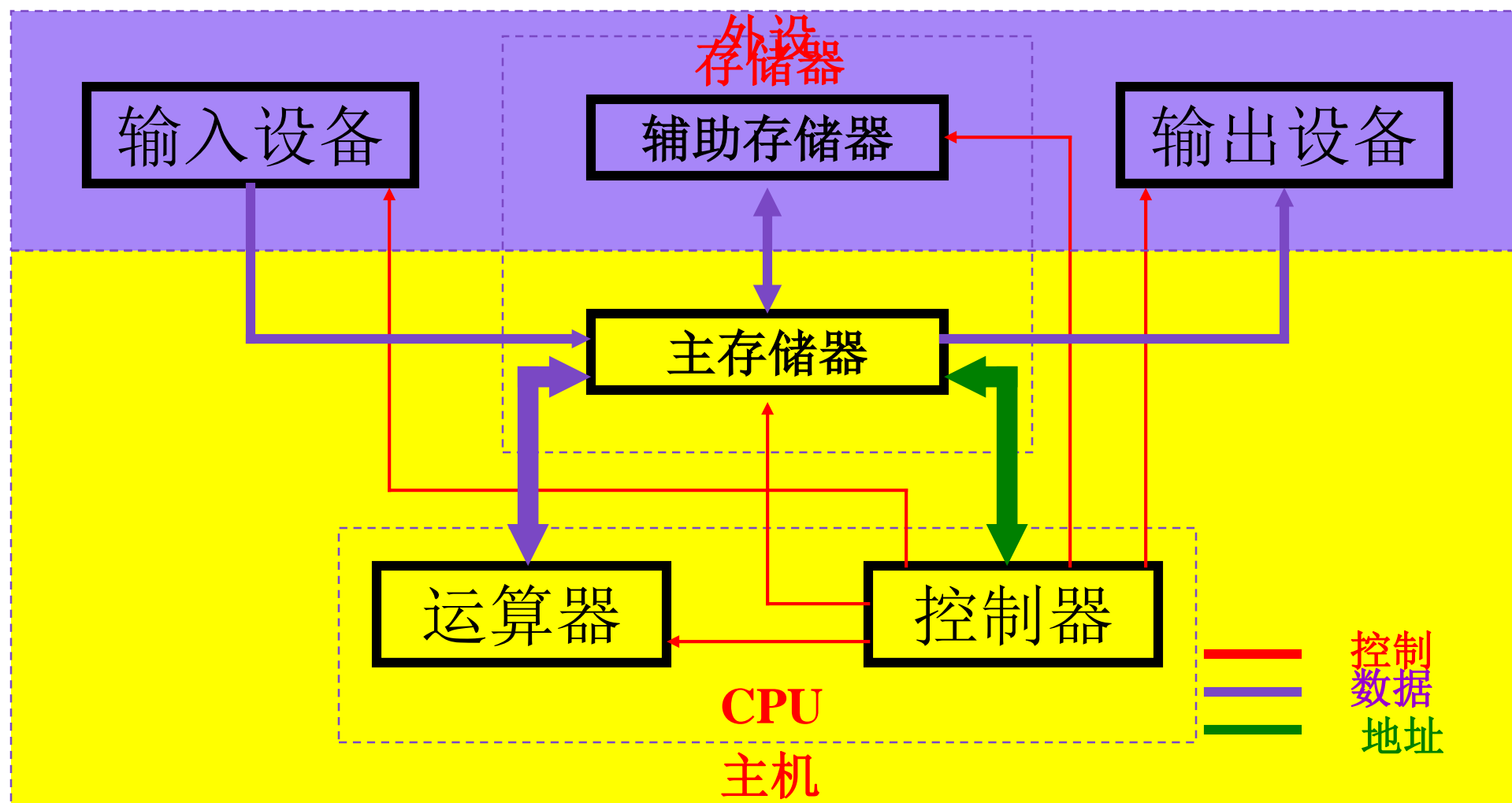
硬件与软件的关系

硬件是计算机系统的物质基础，软件是计算机系统的灵魂。硬件和软件是相辅相成的，不可分割的整体。

当前计算机的硬件和软件正朝着互相渗透，互相融合的方向发展，在计算机系统中没有一条明确的硬件与软件的分界线。硬件和软件之间的界面是浮动的，对于程序设计人员来说，**硬件和软件在逻辑上是等价的。**

固件是指那些存储在能永久保存信息的器件（如ROM）中的程序，是具有软件功能的硬件。

注意





中央处理器 (CPU)

CPU = 运算器 + 控制器

主机

主机 = 中央处理器 + 主存储器

外部设备

除去主机以外的硬件装置

系统总线 地址总线、数据总线和控制总线。

- 最早用机器语言编写程序，并记录在纸带或卡片上

穿孔表示0，未穿孔表示1

输入：按钮、开关；所有信息都是0/1序列！
输出：指示灯等

假设：0010-jxx 转移指令

0: 0101 0110

1: 0010 0100

2:

3:

4: 0110 0111

5:

6:

太原始了，无法忍受，咋办？

用符号表示而不用0/1表示！

若在第4条指令前加入指令，则需重新计算地址码（如jxx的目标地址），然后重新打孔。不灵活！


书写、阅读困难！

- 用助记符表示操作码
- 用标号表示位置
- 用助记符表示寄存器
-

汇编程序将汇编语言转换为机器语言！

与机器指令一一对应。

0:	0101 0110	sub B
1:	0010 0100	jnz L0
2:
3:
4:	0110 0111	L0: add C
5:
6:	B:
7:	C:



汇编语言编写的优点是：

不会因为增减指令而需要修改其他指令

不需记忆指令码，编写方便

可读性比机器语言强

在第4条指令
前加指令时
不用改变sub、
jnz和add指
令中的地址
码！

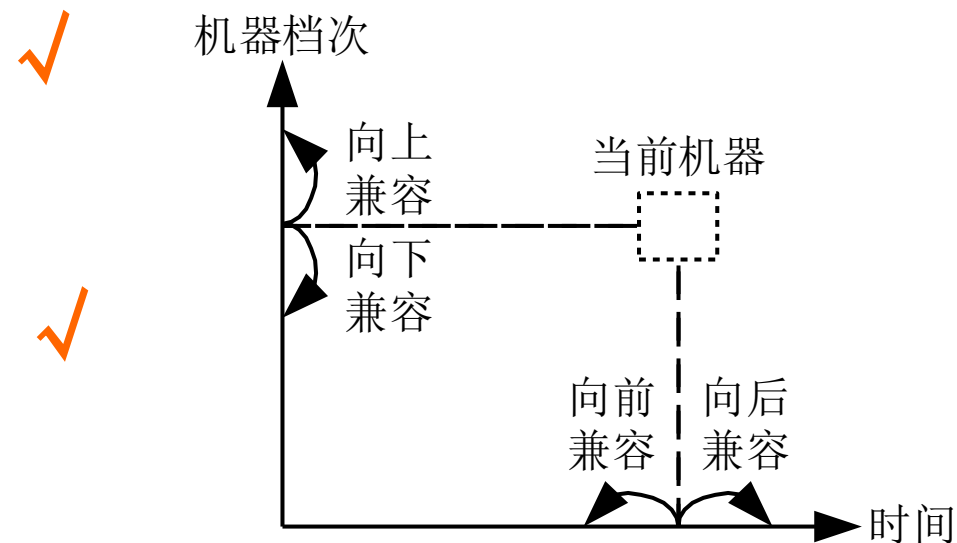
- 随着技术的发展，出现了许多高级编程语言
 - 它们与具体机器结构无关
 - 面向算法描述，比机器级语言描述能力强得多
 - 高级语言中一条语句对应几条、几十条甚至几百条指令
 - 有“面向过程”和“面向对象”的语言之分
 - 处理逻辑分为三种结构
 - 顺序结构、选择结构、循环结构
- 有两种转换方式：“编译”和“解释”
 - 编译程序(Compiler): 将高级语言源程序转换为机器级目标程序，执行时只要启动目标程序即可
 - 解释程序(Interpreter): 将高级语言语句逐条翻译成机器指令并立即执行，不生成目标文件。

- **System software(系统软件)** - 简化编程过程, 并使系统资源被有效利用
 - 操作系统 (Operating System) : 用户接口, 资源管理, ...
 - 语言处理系统: 翻译程序+ Linker, Debug, etc ...
 - 翻译程序(Translator)有三类:
 - 汇编程序(Assembler):** 汇编语言源程序→机器语言目标程序
 - 编译程序(Compiler):** 高级语言源程序→机器级目标程序
 - 解释程序(Interpreter):** 将高级语言语句逐条翻译成机器指令并立即执行, 不生成目标文件。
 - 其他实用程序: 如: 磁盘碎片整理程序、备份程序等
- **Application software(应用软件)** - 解决具体应用问题/完成具体应用任务
 - 各类媒体处理程序: Word/ Image/ Graphics/...
 - 管理信息系统 (MIS)
 - Game, ...

系列机是指一个厂家生产的，具有相同的系统结构，但具有不同组成和实现的一系列不同型号的机器。

系列机应在指令系统、数据格式、字符编码、中断系统、控制方式、输入/输出操作方式等方面保持统一，从而保证软件的兼容性。

软件兼容：
向上兼容
向下兼容
向前兼容
向后兼容



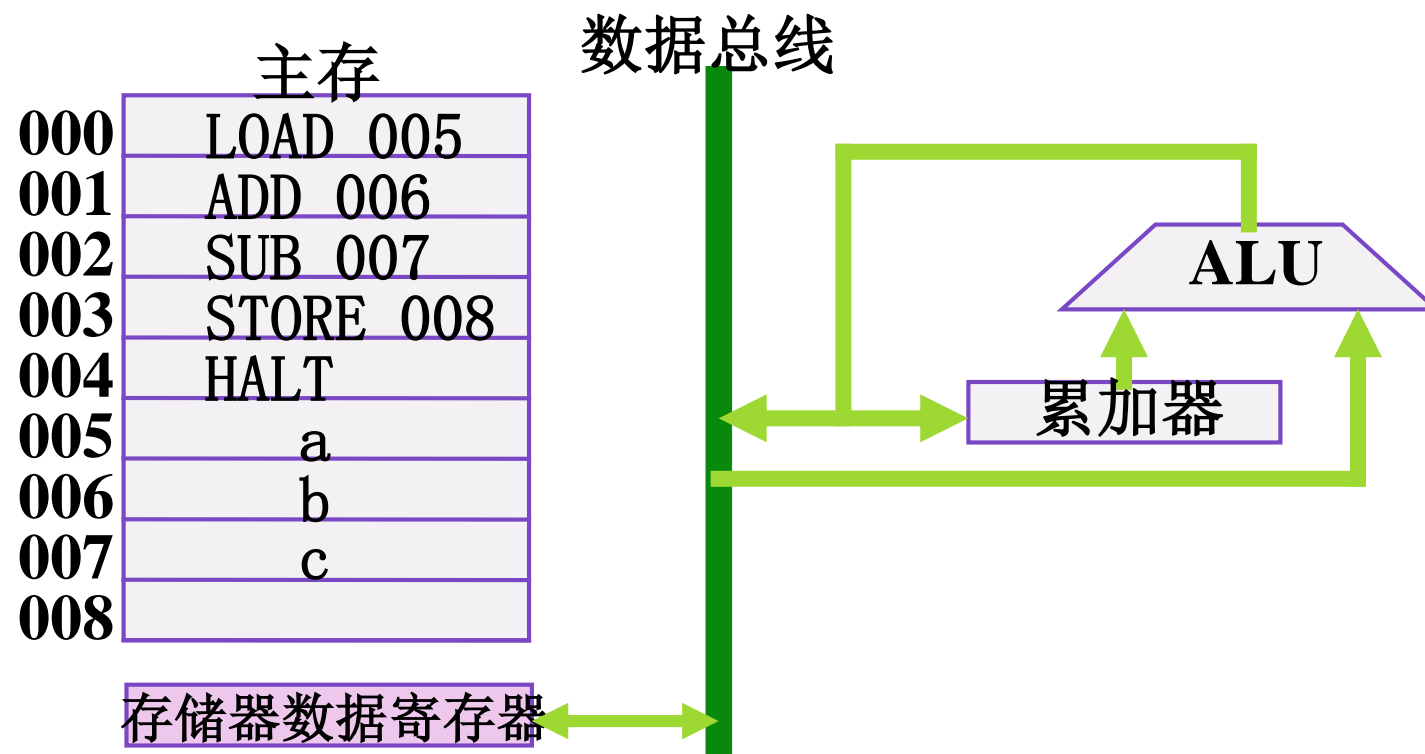
1. 存储程序与计算机系统组成

2. 计算机的工作过程与计算机性能指标

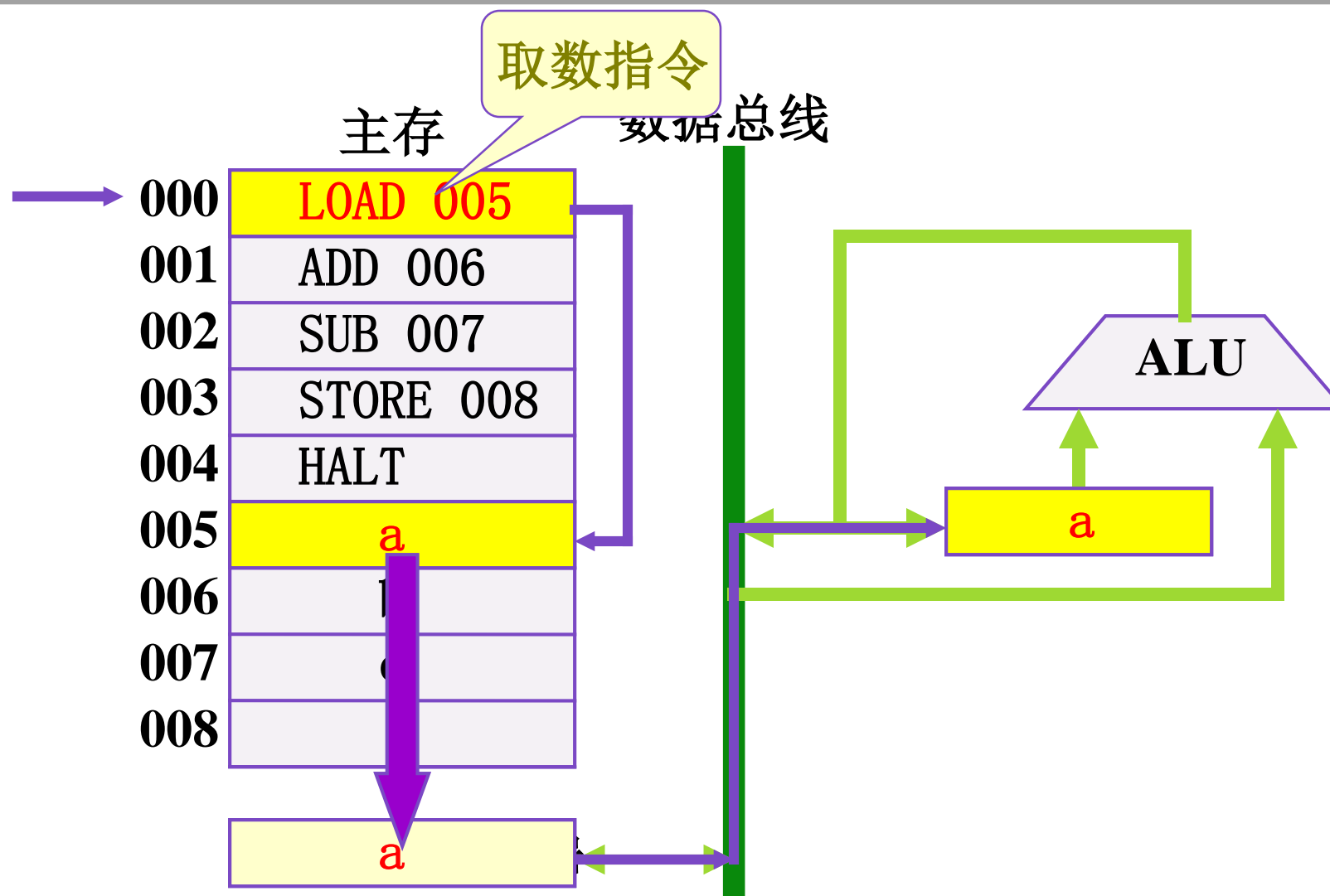
3. 计算机系统发展历程

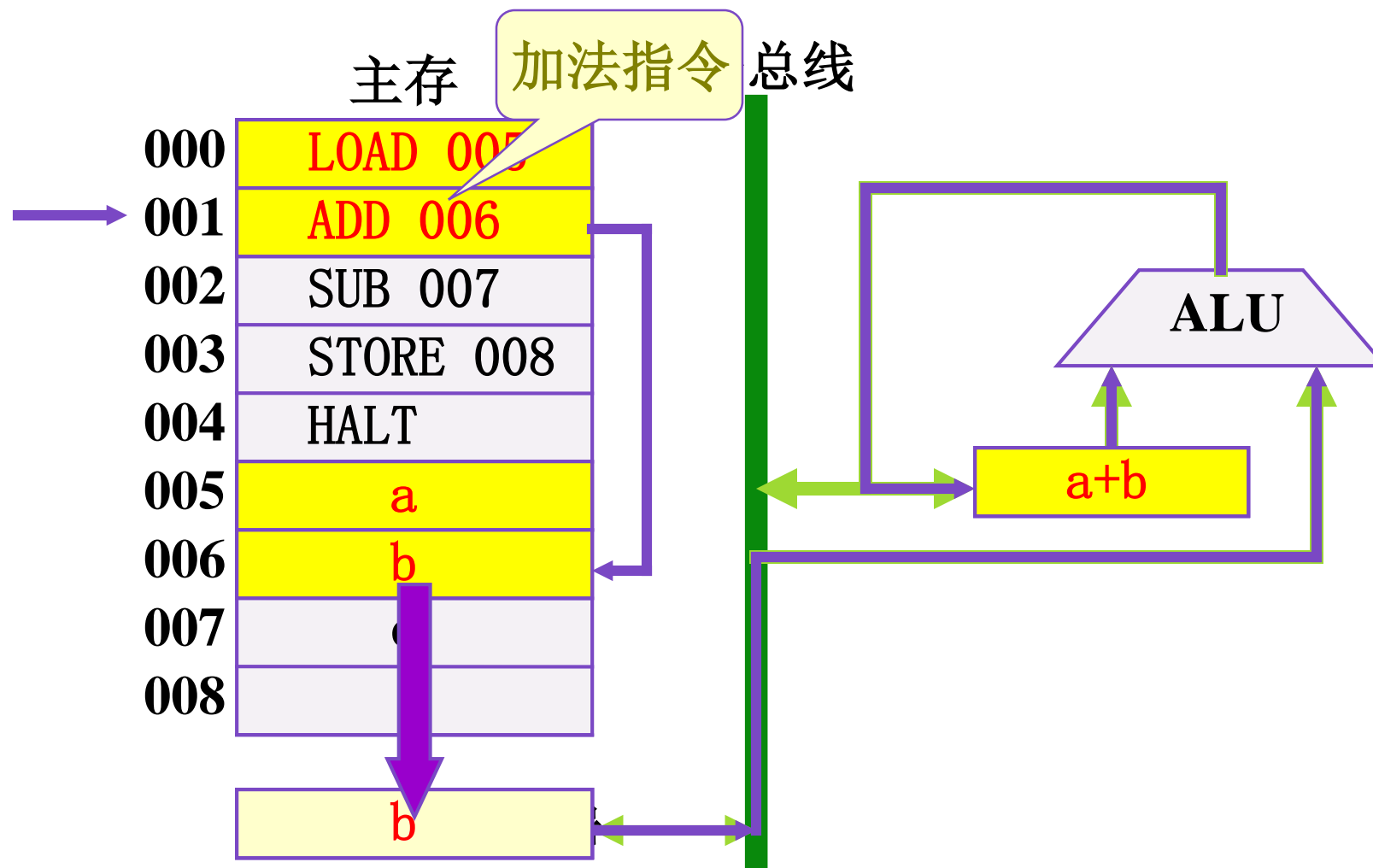
将编制好的程序放在主存中，由控制器控制逐条取出指令执行，以计算 $a+b-c=?$ 为例加以说明。

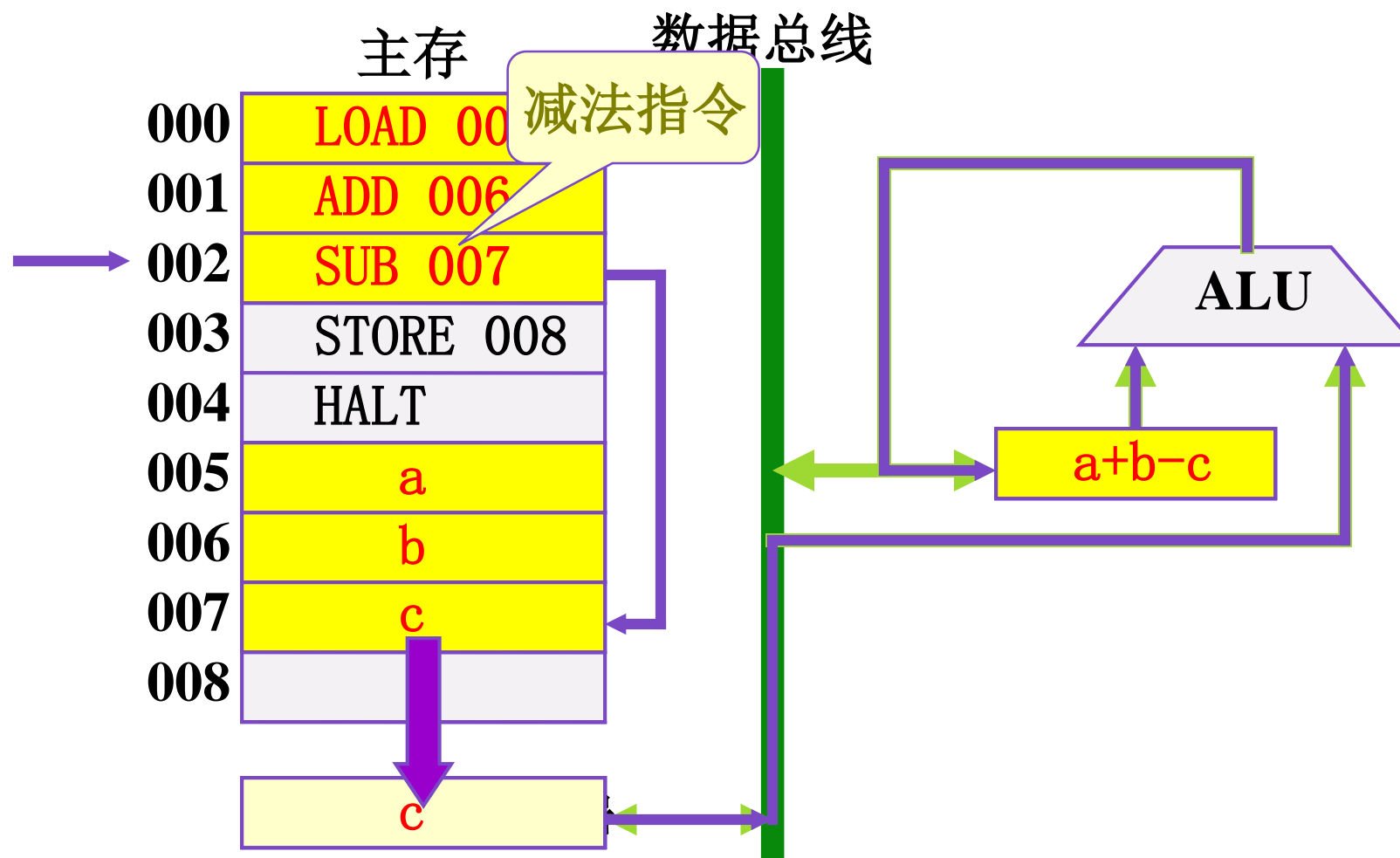
设 a 、 b 、 c 为已知的3个数，分别存放在主存的5~7号单元中，结果将存放在主存的8号单元。

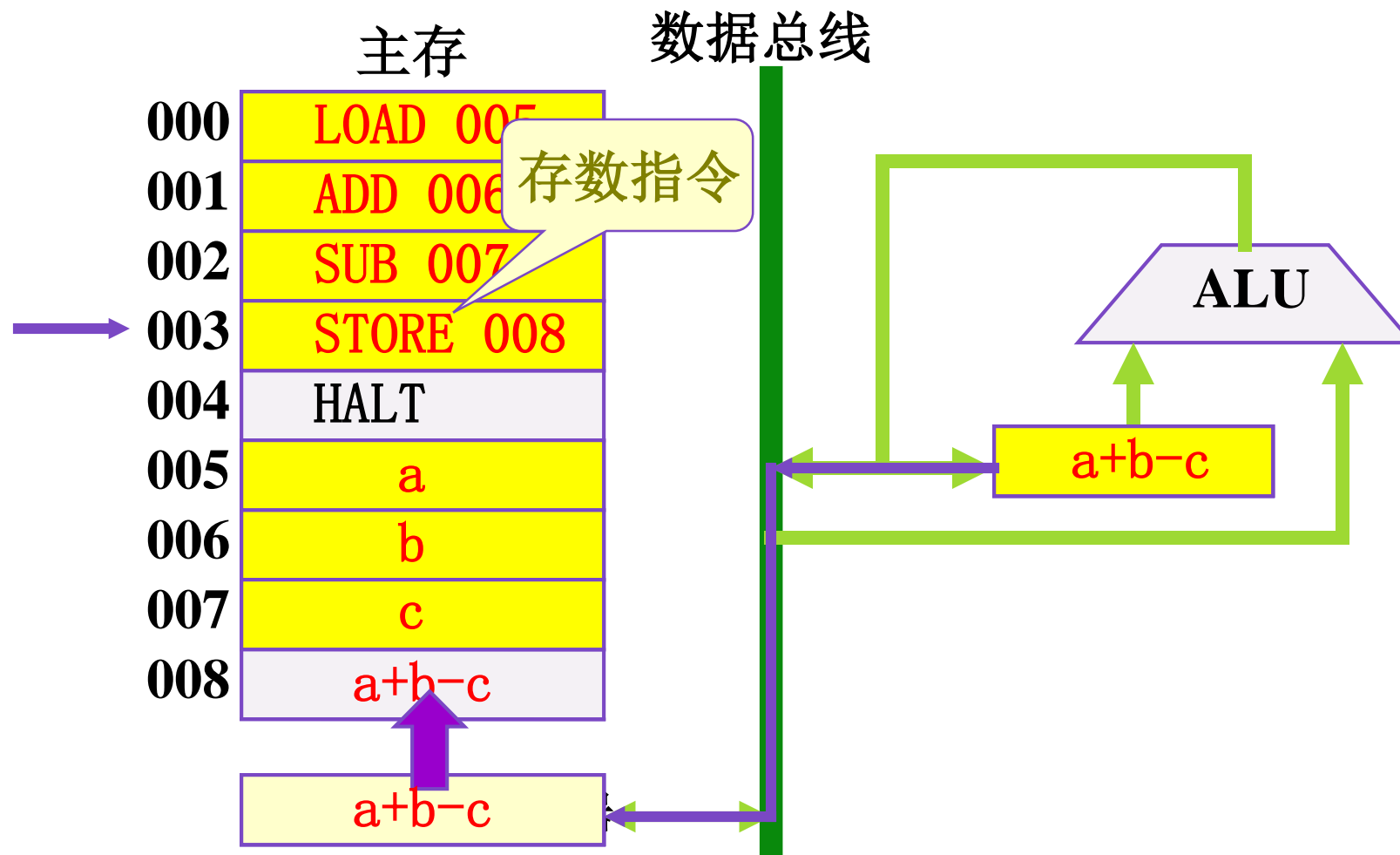


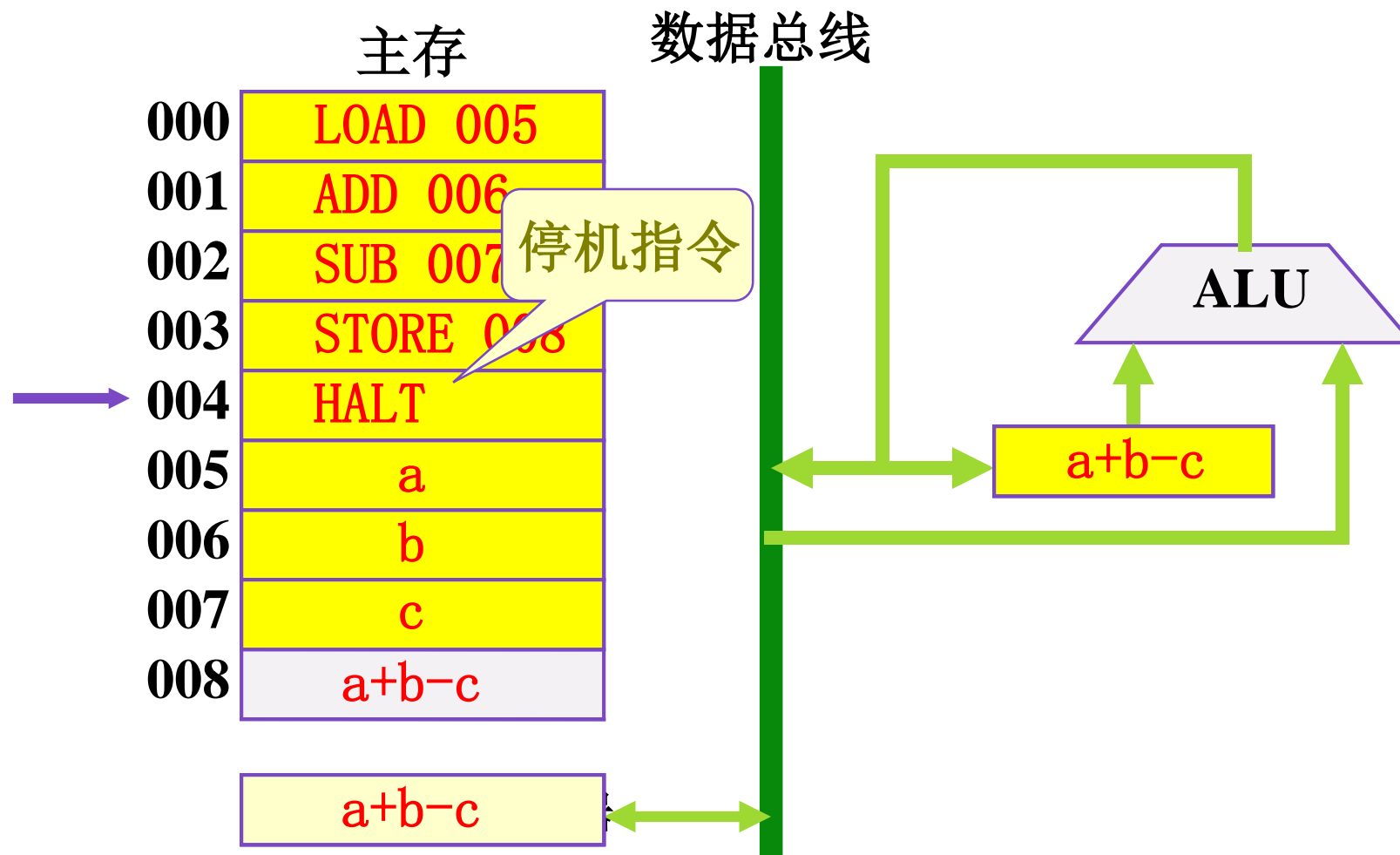
计算机的工作过程











1. 机器字长

机器字长是指参与运算的数的基本位数，它是由加法器、寄存器、数据总线的位数决定的。

在计算机中为了更灵活地表达和处理信息，许多计算机又以**字节 (Byte)** 为基本单位，一个字节等于8位二进制**位 (bit)**。

不同的计算机，字 (Word) 可以不相同，但对于系列机来说，在同一系列中，字却是固定的，如80X86系列中，一个字等于16位；IBM303X系列中，一个字等于32位。

2.数据通路宽度

数据总线一次所能并行传送信息的位数，称为数据通路宽度。它影响到信息的传送能力，从而影响计算机的有效处理速度。这里所说的数据通路宽度是指外部数据总线的宽度，它与CPU内部的数据总线宽度（内部寄存器的大小）有可能不同。

内、外数据通路宽度相等的CPU有：Intel 8086、80286、80486等；

外部 < 内部的CPU有：8088、80386SX等；

外部 > 内部的CPU有：Pentium等。

3.主存容量

一个主存储器所能存储的全部信息量称为主存容量。衡量主存容量单位有两种：

- ① **字节数**。这类计算机称为**字节编址**的计算机。每1024个字节称为1K字节 ($2^{10}=1K$)，每1024K字节称为1M字节 ($2^{20}=1M$)，每1024M字节称为1G字节 ($2^{30}=1G$)，每1024G字节称为1T字节 ($2^{40}=1T$)。
- ② **字数×字长**。这类计算机称为**字编址**的计算机。如：4096×16表示存储器有4096个存储单元，每个存储单元字长为16位。

4.运算速度

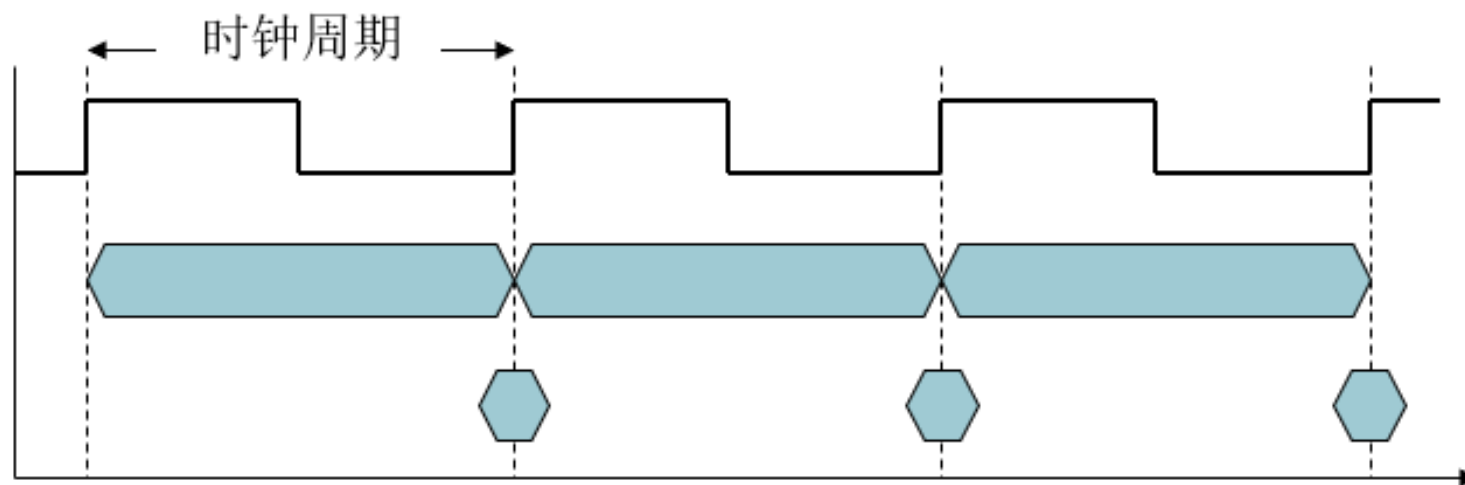
(1)吞吐量 and 响应时间

吞吐量是指系统在单位时间内处理请求的数量。响应时间是指系统对请求作出响应的的时间，响应时间包括CPU时间（运行一个程序所花费的时间）与等待时间（用于磁盘访问、存储器访问、I/O操作、操作系统开销等时间）的总和。

(2)主频和CPU时钟周期

主频是衡量CPU速度的重要参数。**CPU的主频又称为时钟频率，表示在CPU内数字脉冲信号振荡的速度，与CPU实际的运算能力并没有直接关系。主频的倒数就是CPU时钟周期，这是CPU中最小的时间元素。每个动作至少需要一个时钟周期。**

时钟（循环）
数据传输
和运算
更新状态



- 时钟周期：一个时钟循环的时间
 - 例如 $250\text{ps} = 0.25\text{ns} = 250 \times 10^{-12}\text{s}$
- 时钟频率（速率）：每秒的周期数
 - 例如 $4.0\text{GHz} = 4000\text{MHz} = 4.0 \times 10^9\text{Hz}$

(3) CPI

CPI (Cycles per Instruction) 就是每条指令执行所用的时钟周期数。由于不同指令的功能不同，造成指令执行时间不同，也即指令执行所用的时钟数不同，所以CPI是一个平均值。在现代高性能计算机中，由于采用各种并行技术，使指令执行高度并行化，常常是一个系统时钟周期内可以处理若干条指令，所以CPI参数经常用**IPC (Instructions per Cycle)** 表示，即每个时钟周期执行的指令数。 $IPC=1/CPI$

若将程序执行过程中所处理的指令数，记为IC。这样可以获得一个与计算机系统结构有关的参数，即“指令时钟数CPI”。

$$CPI = \frac{\text{CPU时钟周期数}}{IC}$$



假设计算机系统有 n 种指令，其中第 i 种指令的处理时间为 CPI_i ，在程序中第 i 种指令出现的次数为 I_i ，则有：

$$CPI = \frac{\sum_{i=1}^n CPI_i \times I_i}{IC} = \sum_{i=1}^n (CPI_i \times \frac{I_i}{IC})$$

- 有两个编译过的代码序列可选，都使用A、B、C三类指令

指令种类	A	B	C
每类指令的CPI	1	2	3
序列1的指令数	2	1	2
序列2的指令数	4	1	1

■ 序列1：指令数 = 5

■ 时钟周期数

$$= 2 \times 1 + 1 \times 2 + 2 \times 3 \\ = 10$$

■ 平均CPI = $10/5 = 2.0$

■ 序列2：指令数 = 6

■ 时钟周期数

$$= 4 \times 1 + 1 \times 2 + 1 \times 3 \\ = 9$$

■ 平均CPI = $9/6 = 1.5$

(4) 程序的CPU的执行时间

$$\begin{aligned}\text{CPU执行时间} &= \frac{\text{CPU时钟周期数}}{\text{时钟频率}} \\ &= \frac{\text{指令数} \times CPI}{\text{时钟频率}}\end{aligned}$$

如何提高
性能?

这个公式通常称为**CPU性能公式**。它取决于三个要素：

- ① **时钟频率**：反映了计算机实现技术、生产工艺和计算机组织。
- ② **CPI**：反映了计算机实现技术、计算机指令系统的结构和组织。
- ③ **IC**：反映了计算机指令级体系结构、算法、编程语言和编译技术。

CPU的时间计算举例



- 计算机A: 周期= 250ps, CPI = 2.0
- 计算机B: 周期= 500ps, CPI = 1.2
- 若ISA相同, 哪台更快? 快多少?

$$\begin{aligned}\text{CPU时间}_A &= \text{指令数} \times \text{CPI}_A \times \text{周期}_A \\ &= 1 \times 2.0 \times 250\text{ps} = 1 \times 500\text{ps}\end{aligned}$$

A更快...

$$\begin{aligned}\text{CPU时间}_B &= \text{指令数} \times \text{CPI}_B \times \text{周期}_B \\ &= 1 \times 1.2 \times 500\text{ps} = 1 \times 600\text{ps}\end{aligned}$$

$$\frac{\text{CPU时间}_B}{\text{CPU时间}_A} = \frac{1 \times 600\text{ps}}{1 \times 500\text{ps}} = 1.2$$

...快这么多

(5) MIPS和MFLOPS

MIPS表示每秒百万条指令。

MFLOPS每秒表示百万次浮点运算。

$$\text{MIPS} = \frac{\text{指令条数}}{\text{执行时间} \times 10^6} = \frac{\text{主频}}{\text{CPI}}$$

$$\text{MFLOPS} = \frac{\text{浮点操作次数}}{\text{执行时间} \times 10^6}$$

更多的指标：GFLOPS、TFLOPS、PFLOPS、EFLOPS甚至ZFLOPS等。

一个MFLOPS等于每秒1百万 ($=10^6$) 次的浮点运算。

一个GFLOPS等于每秒10亿 ($=10^9$) 次的浮点运算。

一个TFLOPS等于每秒1万亿 ($=10^{12}$) 次的浮点运算。

一个PFLOPS等于每秒1千万亿 ($=10^{15}$) 次的浮点运算。

一个EFLOPS等于每秒100亿亿 ($=10^{18}$) 次的浮点运算。

一个ZFLOPS等于每秒10万亿亿 ($=10^{21}$) 次的浮点运算。

举例：若某处理器的时钟频率为500MHz，每4个时钟周期组成一个机器周期，执行一条指令需要3个机器周期，则该处理器的一个机器周期_____ns，CPI=_____，平均执行速度为MIPS。

**解答：时钟周期T等于主频的倒数，即 $T=1/500\text{MHz}$
 $=1/(0.5\times 10^9\text{Hz})=2\text{ ns}$**

机器周期 $=4T=4\times 2\text{ ns}=8\text{ ns}$

$\text{CPI}=3\times 4=12$

**则平均速度为： $f/(\text{CPI})=(500\times 10^6)/12=500/12$
 $\text{MIPS}=41.6\text{ MIPS}\approx 42\text{MIPS}$ 。**

1. 存储程序与计算机系统组成

2. 计算机的工作过程与计算机性能指标

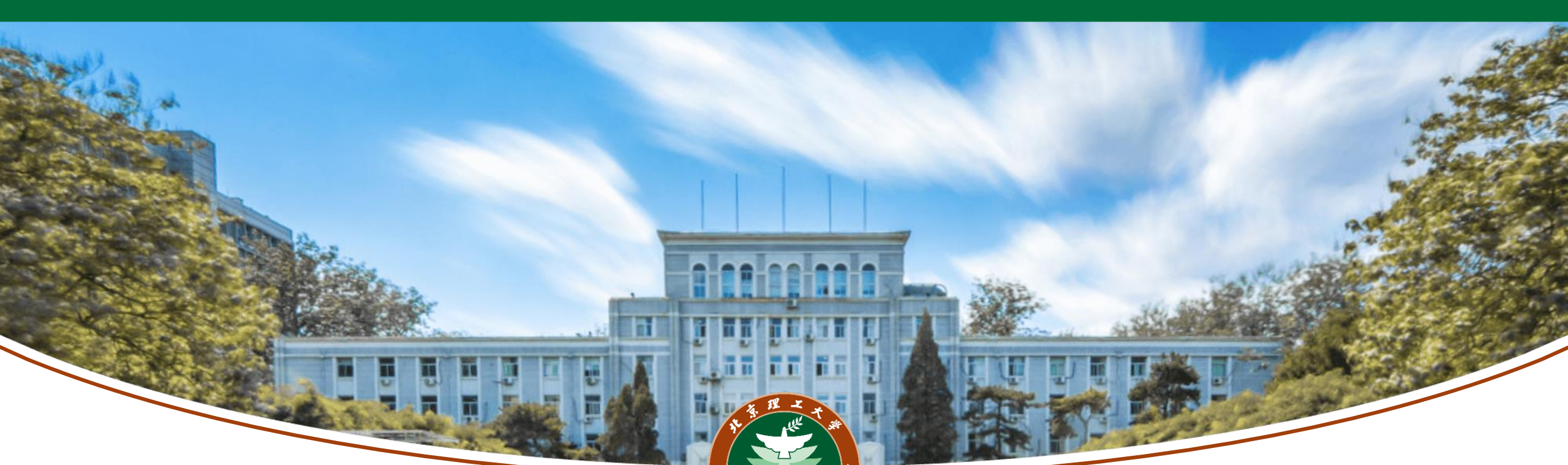
3. 计算机系统发展历程

计算机系统发展历程



最常见的分类方法：按照电子器件来划分的计算机发展史。

类型	时期	主要器件	重要特征
第1代	1946-1957	电子管	速度低，体积大，价格昂贵，可靠性差，主要用于科学计算；
第2代	1958-1964	晶体管	体积缩小，可靠性提高，从科学计算扩大到数据处理；
第3代	1965-1971	中小规模集成电路	体积缩小，可靠性提高，速度达到 MIPS 级，机种多样化，小型计算机出现，软件和外设发展迅速，应用领域扩大；
第4代	1971-	大、超大规模集成电路	速度高达 GIPS 乃至 TIPS 级，多机系统和计算机网络迅速发展，微型计算机出现；



感谢聆听