《微型计算机原理与接口技术》
第6版

第1章 绪 论

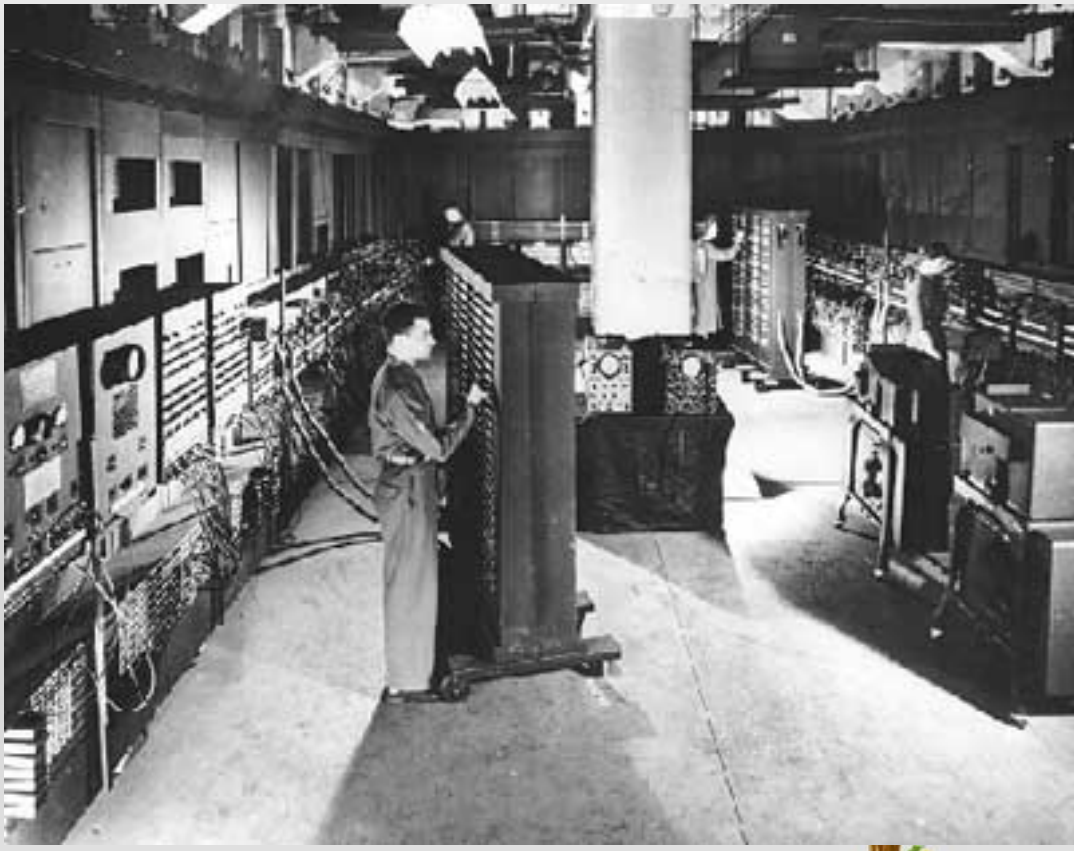
§1.2 计算机的基本结构

历史

- 1946年，美国宾夕法尼亚大学研制成功第一台通用可编程计算机ENIAC（Electronic Numerical Integrator And Calculator）
  - 17000个电子管
  - 500英里导线
  - 重量超过30吨
  - 运算速度10万次/秒


早期的电子管

电子管的功耗大，寿命低，维护难。

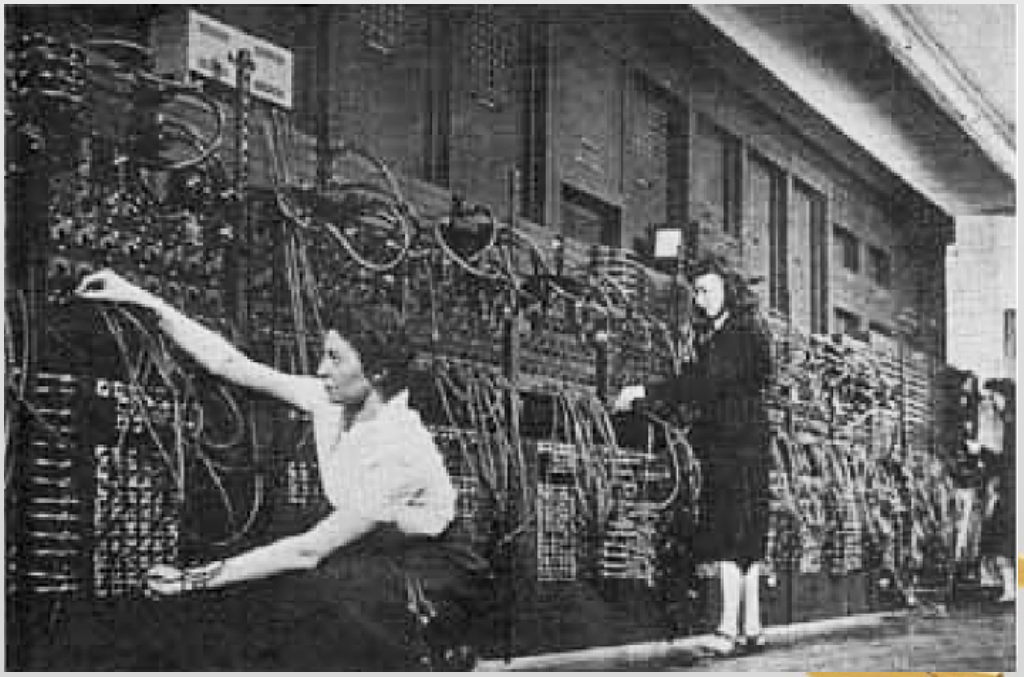


◆历史

- ENIAC推动世界进入了电子计算机时代。

- 编程方法：重新连接线路来实现编程。

  许多工人化几天，对6000多个开关定位，再用转插线连接各控制部件以构成程序序列，很像电话总机的接线。



历史

- **后来采用机器语言（Machine Language）编程**
  - 由1和0组成的代码构成指令（Instruction），告诉计算机要执行的运算和操作。
  - 提高了编程的效率，但用到很多代码，仍很费时。
- **冯·□依曼结构计算机**
  - 数学家冯·□依曼（John Von Neumann）开发出了能接收指令，并将指令保存在存储器中的系统。
  - 为纪念他，常将计算机称为冯·□依曼结构的机器。
  - 半个多世纪以来，计算机技术不断发展，相继出现了各种类型的计算机，就其结构而言，都是冯诺依曼计算机结构的延续和发展。

§1.2 计算机的基本结构

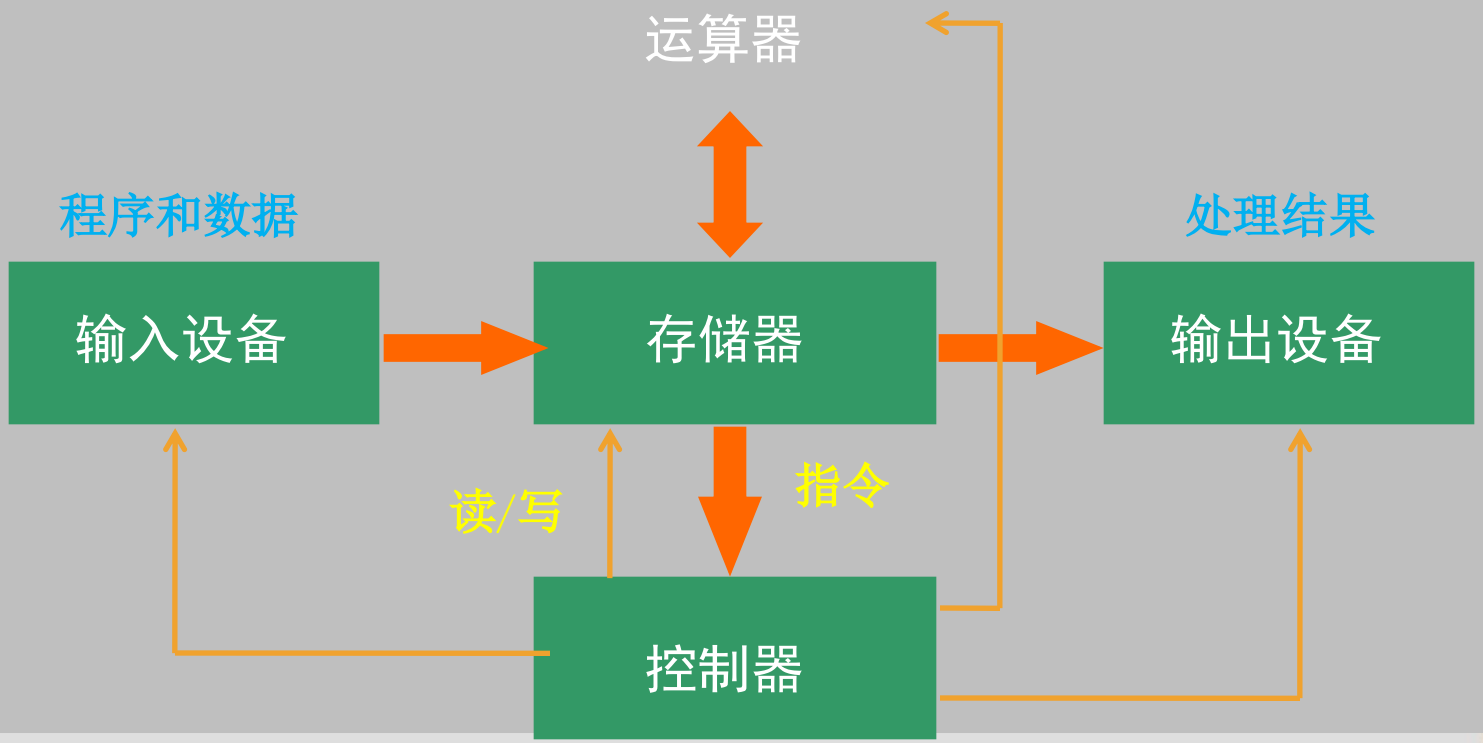
1.2.1 计算机的基本结构

1.2.2 计算机软件

1.2.1 计算机的基本结构

1. 计算机的基本组成

冯.诺依曼计算机的基本框图，包含5个部分：



1. 计算机的基本组成

- **存储器（Memory）**

  以二进制形式存放原始数据、中间结果和程序。

- **运算器（Arithmetic Unit）**

  执行算术运算（＋－×÷）、逻辑运算（与、或、非、异或）和移位等操作的部件，包含：

  加法器或算术逻辑单元（Arithmetic Logic Unit，ALU），累加器（Accumulator）。

- **控制器（Control Unit）**

  指挥和控制各部件协调工作，例如取指令，译码，形成控制命令，让计算机按程序设定的步骤自动操作。

1. 计算机的基本组成

- 输入设备（Input Device）
  - 输入原始数据和程序，转换成计算机能识别的信息，送入存储器去等待处理。
  - 早期的输入设备只有纸带读入机和电传。
- 输出设备（Output Device）
  - 输出运算结果。
  - 打印机是常用的输出设备，后来又发明了显示器、磁带机和磁盘等。

1. 计算机的基本组成

- 运算器和控制器称为中央处理单元（Central Processing Unit，CPU）
- CPU+存储器称为主机
- 输入设备+输出设备称为外部设备（外设）（Peripherals）或I/O设备

1. 计算机的基本组成

- CPU由门电路、寄存器和触发器等高速电子电路组成，经历了电子管、晶体管、集成电路(IC)、大规模集成电路(LSI)和超大规模集成电路(VLSI)等几代。
- IC技术发展, 把整个CPU做在一块芯片上，称为微处理器（Microprocessor），习惯称为CPU。
- 典型微处理器：Intel:8086、80286、80386、80486、Pentium等，Zilog:Z80、Z8000等。
- 用微处理器设计的计算机称为微型计算机（Micro-computer）。
- 早期的微型计算机，如1980年代初推出的IBM PC机以8086/8088为CPU。由于速度较低，外设种类较少，处理能力有限，主要处理个人事务，故称之为个人计算机（Personal Computer，PC）。

2.计算机的工作过程

1）上机前，先把要求解的问题分解为计算机能执行的基本运算，编写好程序，程序由一条条指令组成。

2）将编写好的程序和要处理的原始数据，通过输入设备送到计算机的存储器中存放好。

每个存储单元有一个编号，称之为地址，指令和数据按一定的顺序存放在存储器中。

2.计算机的工作过程

3）启动计算机执行程序。即从程序指定的存储器地址开始逐条取出指令，送到控制器，经译码后产生各种控制信号，送到其它部件，自动执行指令规定的操作。

- 控制器可以向存储器发读/写命令，允许从存储器中取出数据（读），送往运算器进行运算，也可以将运算结果或中间结果送回存储器的指定单元（写），运算完成后将最终结果送到输出设备。
- 控制器向运算器发出各种操作命令，指挥它完成算术运算或逻辑运算等操作。
- 控制器还可向输入或输出设备发启动或停止等命令。

2.计算机的工作过程

4）计算机执行完一条指令后，会自动指向下一条指令的地址，继续取出下一条指令，经译码分析后执行，直至遇到程序中的停机指令后才停止操作。

- 仅有CPU、存储器、外设等硬件（Hardware）构成的计算机称为“裸机”。
- 裸机什么也不会做，必须有指令和程序等软件（Software）配合，才能按设定步骤快速、自动地执行希望的操作。

§1.2 计算机的基本结构

1.2.1 计算机的基本结构

1.2.2 计算机软件

1.2.2 计算机软件

1. 指令和程序

- 程序由一条条指令组成，将它和需要处理的数据一起以二进制的形式送到计算机的存储器中，再启动计算机工作，使机器按这些命令一步步执行。

例如，要让计算机完成操作(a+b)×c，假设a、b、c已存入存储器，就要编写以下指令序列：

- 从存储器中取出a，送到运算器；
- 从存储器中取出b，在运算器中进行(a+b)运算；
- 从存储器中取出c送到运算器；
- 执行(a+b)×c 操作；
- 运算结果送到存储单元，也可输出到外设。

1. 指令和程序

- 指令—用命令形式表示让计算机执行的各种操作。
- 指令集 — 一台计算机所能识别和执行的全部命令称为该机器的指令集（Instruction Set）或指令系统。
  - 不同计算机有不同的指令系统，包含的指令数也不一样。
- 程序—让计算机求解一个数学问题或者完成一项复杂工作前，要把解决问题的过程分解为若干步骤，并用指令序列来表示，以便控制计算机完成这项任务，这样的指令序列就叫程序(Program)。

2.指令的组成和机器码

- 计算机只认得二进制，因此指令都用二进制表示，称为机器码（Machine Code）。

- 指令由操作码（Opecode）和操作数（Operand）组成，操作码说明计算机执行什么操作，操作数指出参加操作的数的本身或操作数所在的地址。

2.指令的组成和机器码

例如，在8086 CPU中，把数字1200H取到累加器AX中去的指令的机器码为：

B8	操作码
00	操作数低字节
12	操作数高字节

- 操作码B8H，规定了要从后面两个字节单元中取出一个2字节数（1200H），送到累加器AX中的操作。
- 数据存放：低字节放在前面，高字节放在后面。
- 指令中的16进制数，在计算机中要存放为二进制。

2.指令的组成和机器码

- 初期，就是用指令的机器码直接来编制程序的，处于机器语言阶段。
- 机器码是一连串的0和1组成的代码，输入计算机时，由纸带穿孔机在纸带上凿孔，有孔表示1，无孔表示0。
- 这种代码不好理解和记忆，还很容易出错，所以编程是一件极其繁杂而困难的工作。

3.汇编语言
Assemble Language

- 汇编语言用助记符（Mnemonic）代替二进制的机器码，用指令功能的英文缩写代替操作码，用符号代替地址，用寄存器存放地址等，用汇编语言程序编程比机器语言方便。

例如，对于8086

数据传送指令用助记符MOV（Move）

加法指令用ADD（Addition）

跳转指令用JMP（Jump）

用RESULT、SUM等符号来表示存储单元地址

将1200H送到累加器AX中去的指令为：

MOV AX，1200H

3.汇编语言

例1.13

编写求解（2+3）=5的汇编语言程序，要求将和存入SUM单元。程序如下：

```
MOV     AX，  2       ;累加器AX←2
ADD     AX，3         ; AX←AX+3
MOV     SUM，AX   ;结果单元SUM←和数5
```

3.汇编语言

- 汇编语言程序比机器语言程序进了一大步。
- 但计算机只认识由0、1组成的机器码。

因此，汇编语言程序必须翻译成机器码表示的目标程序（Object Program），才能被计算机识别和执行。

- 能让计算机自动完成翻译工作的程序称为汇编程序（Assembler）。

3.汇编语言

- 汇编语言的不足之处
  - 汇编语言的语句与机器语言一一对应，因此汇编语言程序语句很多，编程工作很繁琐。
  - 程序员必须十分熟悉CPU的指令系统。
  - 汇编语言的针对性很强，在某种机器上编写的汇编语言程序，不能直接在别的机器上运行。
- 于是，各种高级语言应运而生。

4.高级语言

High-level Programming Language

- 更接近于人们使用习惯的程序设计语言。
  - 允许用英文编写解题的计算程序；
  - 程序中的运算符号和式子，与数学式子类似；
  - 程序员不必了解具体的机器，就能编写出通用性更强的程序。

例如 BASIC、FORTRAN、PASCAL、COBOL、JAVA、C、C++

4.高级语言

- 用高级语言编写的程序，必须翻译成机器指令表示的目标程序，计算机才能执行。因此，需要有各种翻译程序。

例如：

- BASIC用的解释程序（Interpreter）
- FORTRAN、C、COBOL等用的编译程序（Compiler）

4.高级语言

- 高级语言有许多优点，使用极广泛。

  特别是C/C++，允许程序员几乎完全控制程序设计环境和计算机系统，在许多情况下能替代汇编语言。

- 汇编语言在程序设计中仍是重要角色，例如，为PC写的视频游戏程序，几乎都用汇编语言编写。

- 只有对计算机软、硬件了解很透彻的高水平人员，才能熟练地用汇编语言编程。

5.操作系统
Operating System

- 早期计算机既无键盘、显示器、磁盘等外设，也无操控程序。用户带着记录有程序和数据的卡片或打过孔的纸带，拨动计算机面板上的开关将程序输入机器运行。
- 计算机技术发展到多道程序能成批自动运行，于是出现了能控制计算机中所有资源（CPU、存储器、I/O设备及各种软件），使多道程序能成批自动运行，且充分发挥各种资源最大效能的操作系统（OS）。

5.操作系统

- 操作系统是计算机中不可缺少的系统软件，它直接控制和管理系统中的软、硬件资源，合理组织工作流程，并提供各种服务功能，使用户能灵活有效地使用计算机。
- 操作系统包括5个方面的功能：

  处理器管理、存储器管理、设备管理、文件管理、作业管理
- 根据使用环境和提供的功能，分为：

  分时操作系统、实时操作系统、网络操作系统、分布式操作系统 等

5.操作系统

- 1973年，比尔•盖茨□□出了第一个BASIC语言解释器，为MS-DOS操作系统奠定了基础，并凭借该项目的版权费创建了Microsoft（微软）公司。
- 1980年代，微软为IBM　PC机开发了第一个磁盘操作系统DOS(Disk Operation System)。
  - DOS使用字符界面，用户从键盘输入命令执行程序；开始是单任务操作系统，同一时刻只能运行一个任务；DOS　4.0具备了初步的多任务管理能力。

5.操作系统

- 1985年微软公司推出了基于图形用户界面的多任务操作系统Windows。
- 随后设计了多个版本的Windows，如Win 95，Win 98、Win NT、Win 2000、Win XP、Win 7、Win8、Win10等。
- 几乎所有微型计算机上都装有Windows操作系统。

5.操作系统

- DOS操作系统已很少使用，但仍有不少应用程序需要在DOS环境下运行。
- 因此，Windows兼容MS-DOS。
- 可在执行“开始”和“运行”命令后，键入“cmd”命令，就能进入DOS命令行，执行DOS命令和运行DOS环境下的程序。

5.操作系统

- UNIX操作系统，1970年就在小型机上运行，用汇编语言编写，3年后改用C语言编写，具有在不同CPU平台上运行的可移植性。
- 之后又开发出了Linux操作系统，它是一套可免费使用和自由传播的UNIX操作系统，用在Intel　X86系列的计算机上。它由全世界成千上万的程序员设计和实现，能为PC机用户提供UNIX的全部特性。