



北京航空航天大学计算机学院

School of Computer Science and Engineering, Beihang University

计算机组成 (2020级)

计算机组成课程组

(高小鹏、刘旭东、肖利民、牛建伟、万寒)

Tel : 82316285

Mail: liuxd@buaa.edu.cn

liuxd@act.buaa.edu.cn

第一部分：概述

一. 计算机组成与结构简介

1. 计算机的基本组成
2. 计算机的层次结构

二. 计算机中数据信息的表示

1. 无符号数和有符号数
2. 定点数、浮点数表示
3. 非数值数据的表示

三. 计算机的基本工作过程

1. 指令的含义
2. 程序的执行

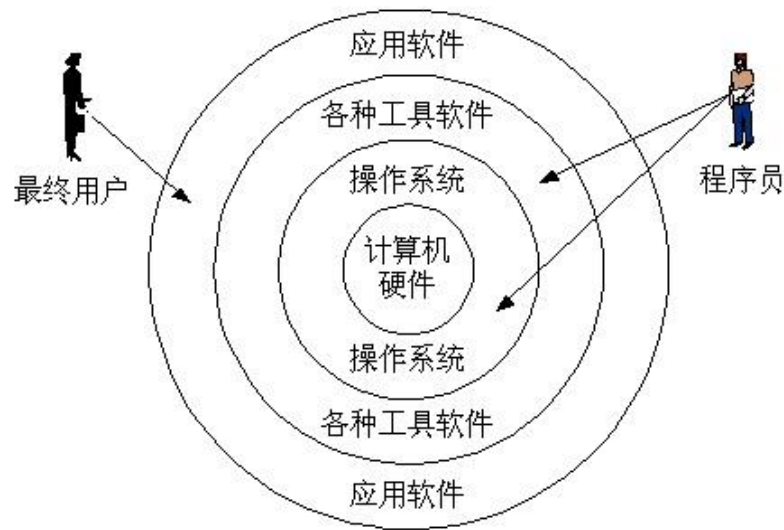
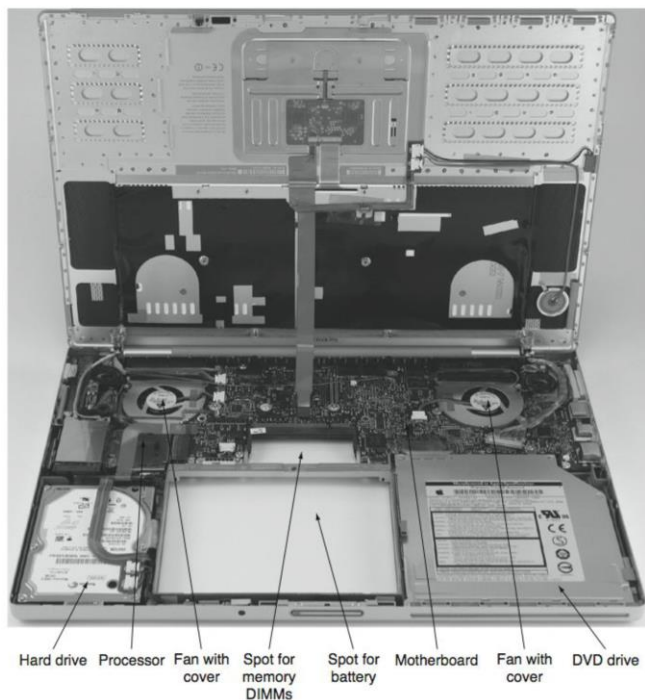
1.1 计算机的基本组成

❖ 硬件 (Hardware)

➤ 计算机的物理部分，可以实现计算机最基本的操作行为。

❖ 软件 (Software)

➤ 使计算机实现各种功能的程序集合。包括系统软件、应用软件两大类。

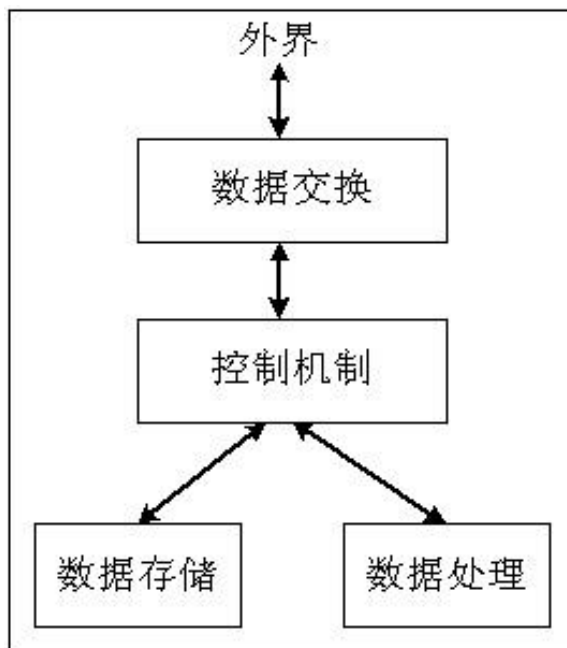


1.1 计算机的基本组成

❖ 计算机的功能

- Data Processing (数据处理)
- Data Storage (数据存储)
- Data Movement (数据移动, 交换)
- Control (控制)

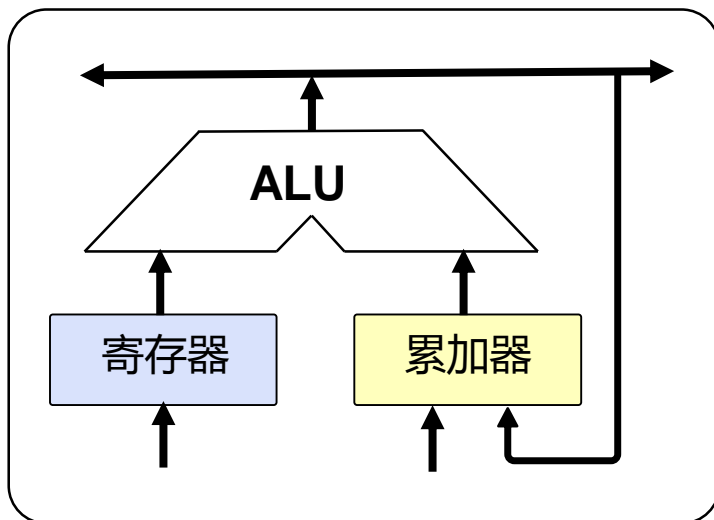
❖ 计算机的功能结构



1.1 计算机的基本组成

❖ 运算器：实现数据处理的部件

- 完成最基本的算术逻辑运算（加、减、乘、除、与、或、非等）
- 算术逻辑单元ALU (Arithmetic and Logic Unit)
- 累加器、寄存器部件



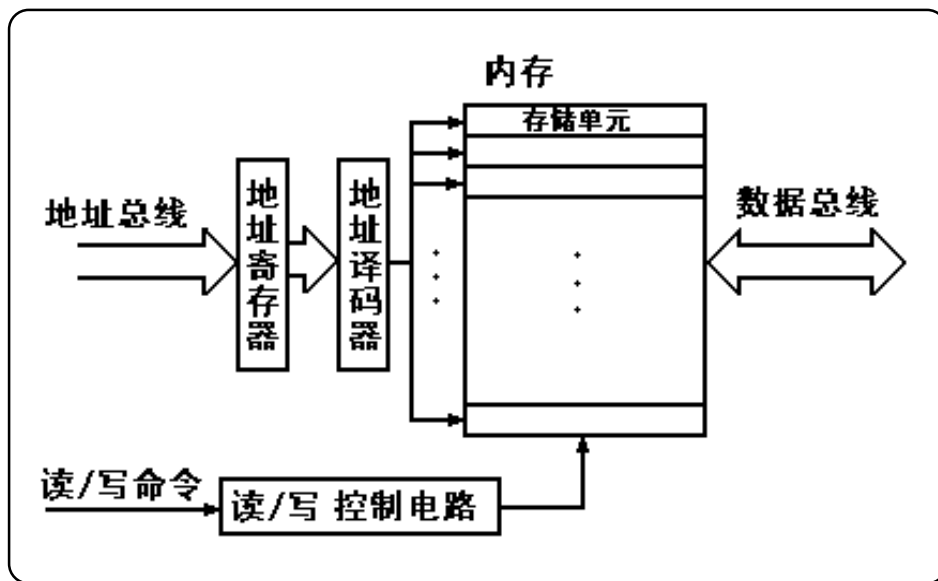
运算器结构示意图

机器字长：运算器一次运算所处理的二进制数的位数。常见的机器字长有16位、32位、64位。

1.1 计算机的基本组成

❖ 存储器：实现数据存储的部件

- 存放程序和数据（二进制信息）
- 存储单元：字节（Byte，8位），字（Word，位数为机器字长）
- 地址：每一个存储单元拥有一个唯一的编号，称为地址
- 存储器的工作方式：读、写



存储器结构示意图

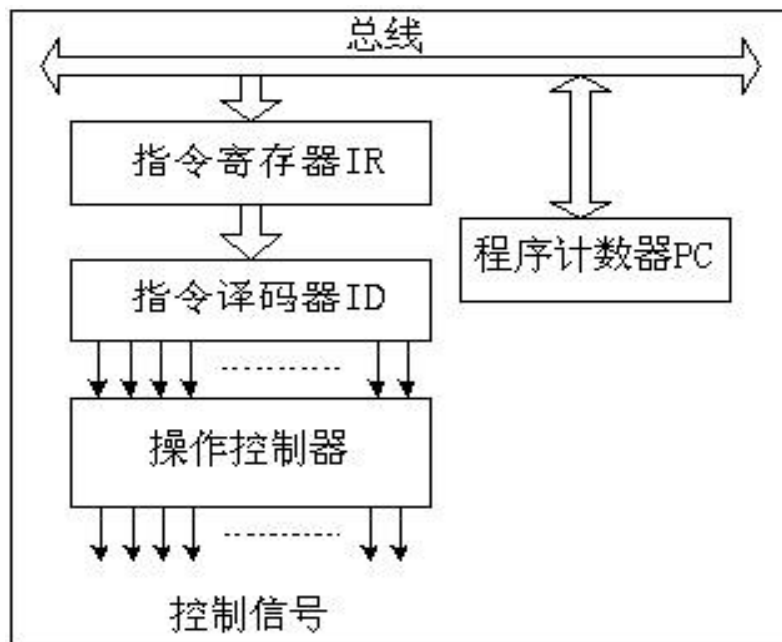
地址总线：传送访问存储器的地址信息，位数与存储器的容量相关，地址总线是单向总线。

数据总线：传送读存储器所读取的数据或写存储器所要写入的数据，数据总线是双向总线。

1.1 计算机的基本组成

❖ 控制器：实现控制功能的部件

- 解释计算机指令，产生各功能部件完成指令所需的控制信号，控制功能部件协同工作
- 指令计数器（Program Counter）：保存指令所在存储单元的地址
- 指令寄存器（Instruction Register）：保存指令二进制代码
- 指令译码器（Instruction Decoder）：解释指令
- 操作控制器：根据指令译码结果产生各功能部件所需的操作控制信号
- 时序部件：提供时钟信号和时序控制信号



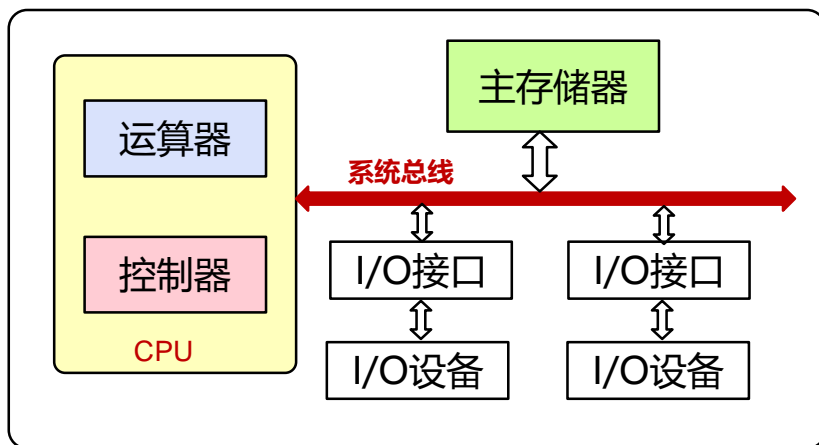
控制器结构示意图

中央处理器CPU：一般将控制器+运算器统称为CPU。

1.1 计算机的基本组成

❖ 输入输出：实现数据交换的部件

- 实现计算机内部与外界（其他系统或人类）的信息交换
- 实现数据交换的设备：输入设备、输出设备
- I/O接口：实现输入/输出设备与计算机系统总线的信息交互



总线 (Bus)：连接两个或多个功能部件的公共信息通路。系统总线一般包括地址线、数据线和控制线。

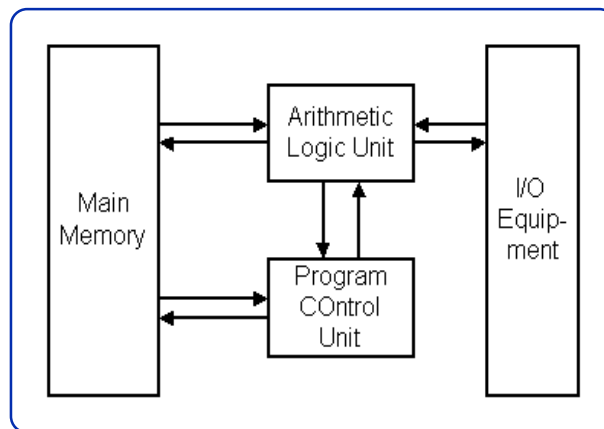
1.1 计算机的基本组成

❖ **冯·诺依曼结构：**1946年，冯·诺依曼与同事开始研制 IAS。该机结构被公认为随后发展起来的通用计算机的结构原型。

- 计算机由运算器、存储器、控制器和输入输出部分组成
- 指令和数据用二进制表示，两者在形式上没有差别
- 指令和数据存放在存储器中，按地址访问
- 指令有操作码和地址码两个部分组成，操作码指定操作性质，地址码指定操作数位置
- 采用“存储程序”方式进行工作

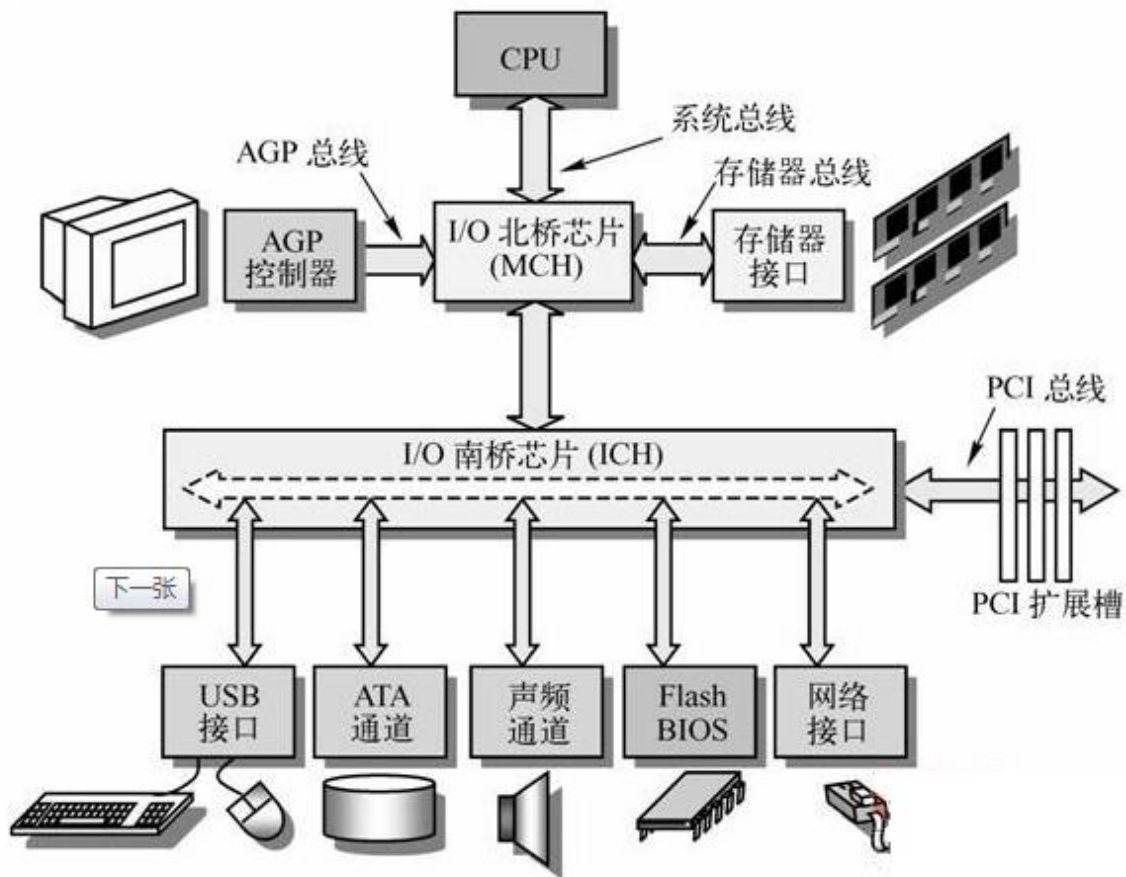


冯·诺依曼



IAS结构

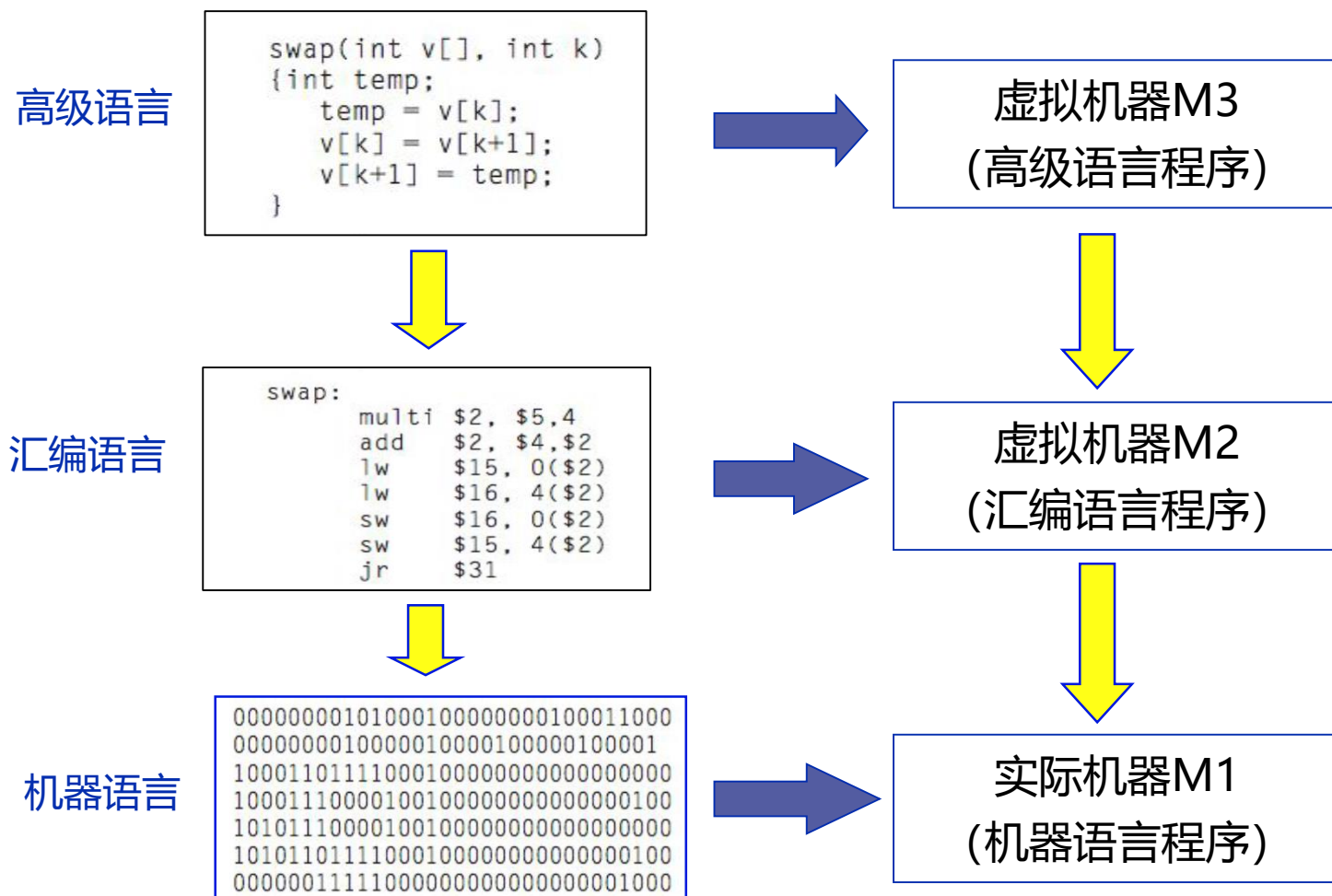
1.1 计算机的基本组成



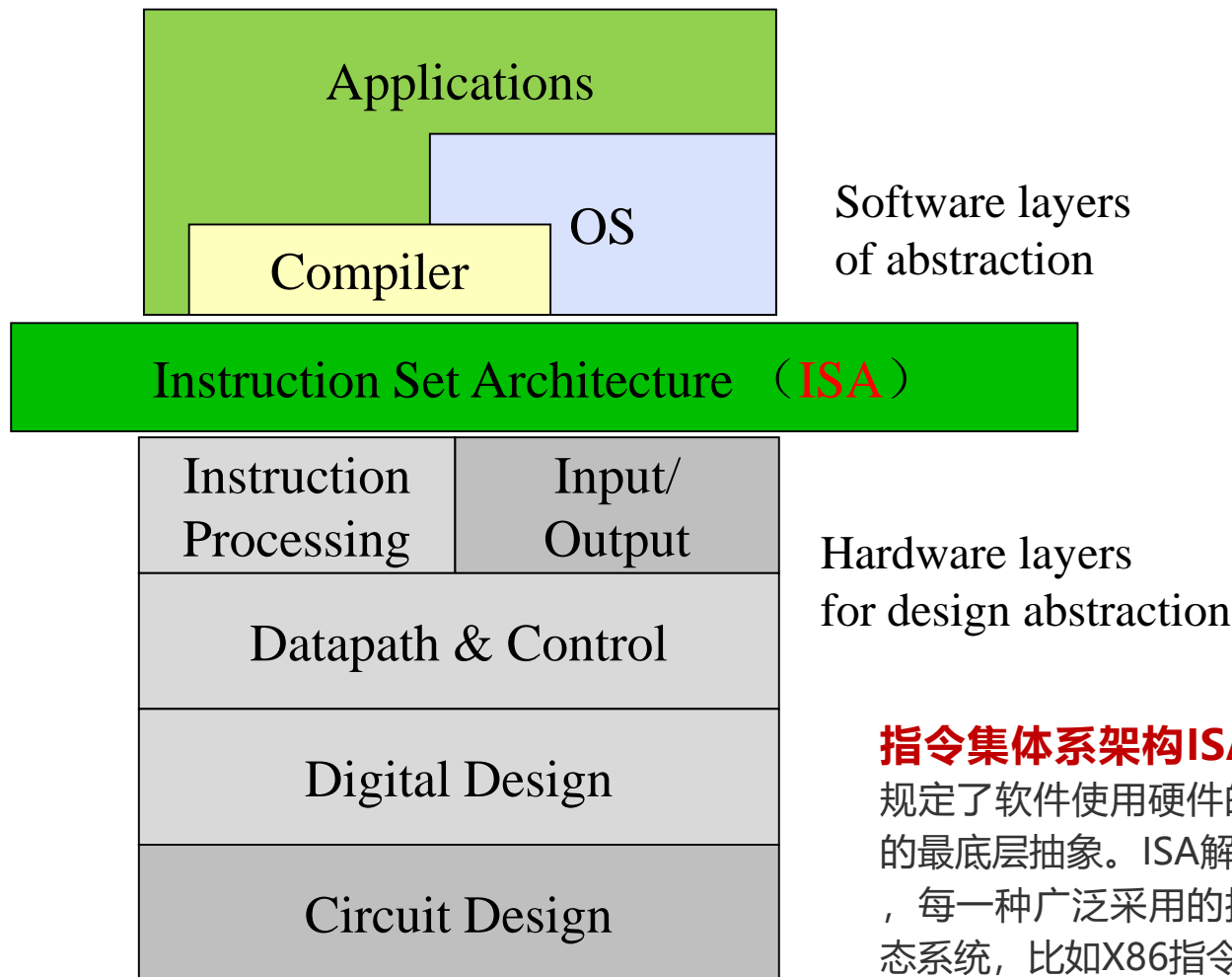
普通PC的内部结构（多总线结构）

1.2 计算机系统层次结构

❖ 计算机的层次结构的演变



1.2 计算机系统层次结构



指令集体系架构ISA: 一种规约(Specification), 规定了软件使用硬件的细节, 是对硬件结构和功能的最底层抽象。ISA解决了最基本的软件兼容性问题, 每一种广泛采用的指令集背后都有一个强大的生态系统, 比如X86指令集架构和ARM指令集架构。

- 1. 机器字长:** 指CPU一次处理的二进制数据的位数，一般与计算机内部的运算器、寄存器以及数据总线的宽度一致；
- 2. 主存容量:** 主存储器信息存储量，一般以字节为单位。1KB(kilobyte)= 2^{10} 字节，1MB(Megabyte)= 2^{20} 字节，1GB(Gigabyte)= 2^{30} 字节，1TB(terabyte)= 2^{40} 字节，1PB(Petabyte)= 2^{50} 字节；
- 3. 时钟周期:** 计算机最小时间单位。一个时钟周期内，CPU仅能完成一个最基本的操作；
- 4. CPI(Clock Cycles Per Instruction):** 计算机执行每条指令所需要的平均时钟周期数；
- 5. MIPS(Million Instructions Per Second):** 每秒执行的百万条机器指令数，是衡量计算机性能（速度）的一个指标，更大的单位有GIPS(Giga Instructions Per Second) ；
- 6. MFLOPS(Million Floating-Point Operations Per Second):** 每秒执行百万浮点运算的次数，是衡量计算机浮点运算性能的指标。

第一部分：概述

一. 计算机组成与结构简介

1. 计算机的基本组成
2. 计算机的层次结构

二. 计算机中数据信息的表示

1. 无符号数和有符号数
2. 定点数、浮点数表示
3. 非数值数据的表示

三. 计算机的基本工作过程

1. 指令的含义
2. 程序的执行

01000001

编码：41H



这是什么？

字符：字母A（ASCII码）

整数：+ 65

小数：+ 0.253906

❖ 计算机数据信息表示的基本问题

- 采用二进制，只有1和0；
- 类型：字符？数值？
- 数的符号：正数、负数、零
- 数的形态：整数、小数、小数点的性质；

2.1 无符号数和有符号数

❖ 无符号数

- 数的编码中所有位均为数值位，没有符号位
- 只能表示 ≥ 0 的正整数
- N位无符号数的表示范围： $0 \sim 2^n - 1$
- 一般在全部是正数运算且不出现负值结果的情况下，可使用无符号数表示，例如地址运算。

1001 1100B

$$= 1 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$$

=156D

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2	4	8	16	32	64	128	256	512	1024	2048	4096	8192	16384	32768	65536

2.1 无符号数和有符号数

❖ 有符号数

- 实例：+ 0.1010110, - 0.1101001, + 1001.001, -1101101

❖ 机器数表示

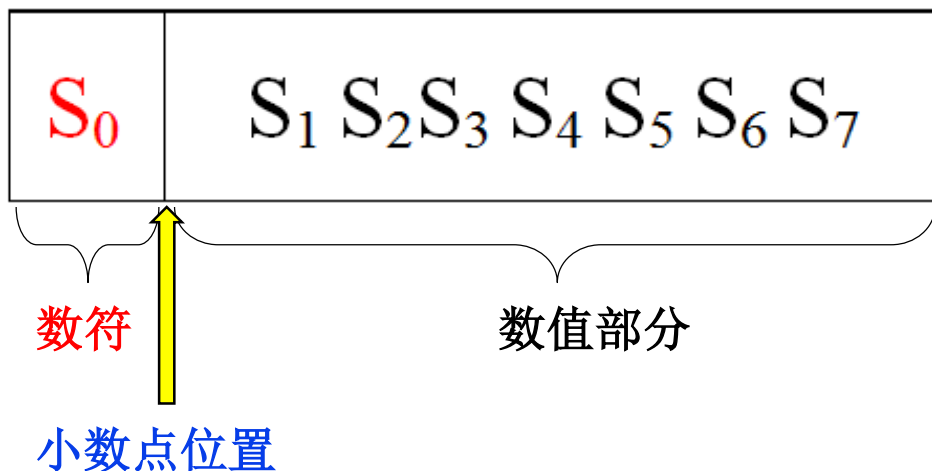
- 数的正负问题：设**符号位**，“0”表示“正”，“1”表示“负”，固定为编码的最高位
- 真值0怎么办：正零，负零
- 小数点怎么办：隐含，小数点位置固定（即**定点数**）
 - **定点小数**：小数点在最高位数值位前，绝对值小于1
 - **定点整数**：小数点在最低位数值位后，没有小数部分
- 带有整数和小数部分的数怎么办：**浮点数**，按2为基的科学表示方法表示

C语言中变量为什么要一定先定义类型才能使用
char、int、unsigned、float、double

2.1 无符号数和有符号数

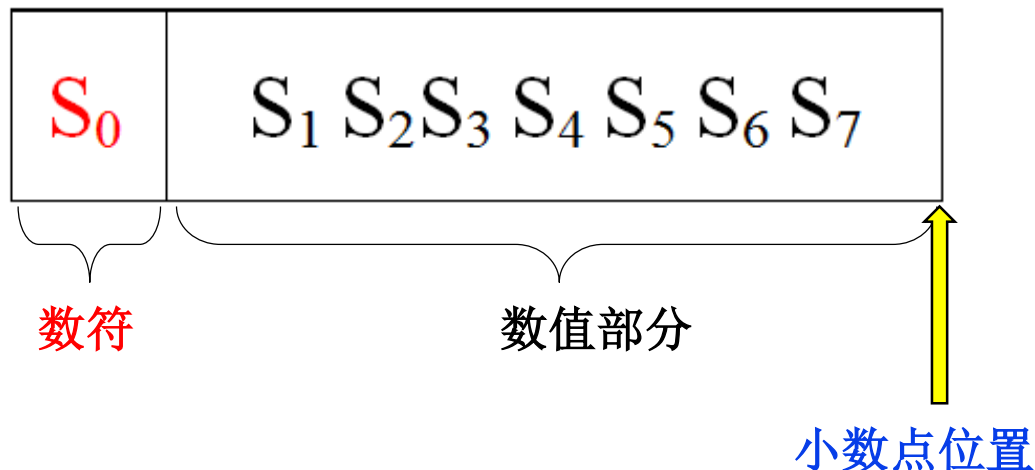
➤ 定点小数

如：01100000
是十进制的0.75



➤ 定点整数

如：01100000
是十进制的192



2.2 定点数表示

❖ n位定点整数表示及范围（原码、反码、补码、移码）

$$[x]_{\text{原}} = \begin{cases} x & 0 \leq x \leq 2^{n-1} - 1 \\ 2^{n-1} - x & -(2^{n-1} - 1) \leq x \leq 0 \end{cases}$$

$$[x]_{\text{反}} = \begin{cases} x & 0 \leq x \leq 2^{n-1} - 1 \\ (2^n - 1) + x & -(2^{n-1} - 1) \leq x \leq 0 \end{cases}$$

$$[x]_{\text{补}} = \begin{cases} x & 0 \leq x \leq 2^{n-1} - 1 \\ 2^n + x & -2^{n-1} \leq x \leq 0 \end{cases}$$

$$[x]_{\text{移}} = 2^{n-1} + x \quad -2^{n-1} \leq x \leq 2^{n-1} - 1$$

Q

1. 8位定点整数原码、反码、补码的表示范围是多少？
2. 对应的二进制表示分别是什么？

2.2 定点数（定点整数为例）

4位二进制机器数编码

十进制数值	原码	反码	补码
0	0000	0000	0000
1	0001	0001	0001
2	0010	0010	0010
3	0011	0011	0011
4	0100	0100	0100
5	0101	0101	0101
6	0110	0110	0110
7	0111	0111	0111
-0	1000	1111	0000
-1	1001	1110	1111
-2	1010	1101	1110
-3	1011	1100	1101
-4	1100	1011	1100
-5	1101	1010	1011
-6	1110	1001	1010
-7	1111	1000	1001
-8	/	/	1000

2.2 定点数表示

❖ n位定点小数表示及范围（原码、反码、补码）

$$[x]_{\text{原}} = \begin{cases} x & 0 \leq x < 1 \\ 1 + |x| & -1 < x \leq 0 \end{cases}$$

$$[x]_{\text{反}} = \begin{cases} x & 0 \leq x < 1 \\ (2 - 2^{-n}) + x & -1 < x \leq 0 \end{cases}$$

$$[x]_{\text{补}} = \begin{cases} x & 0 \leq x < 1 \\ 2 + x & -1 < x \leq 0 \end{cases}$$

2.2 定点数

❖ 原码

- 容易理解
- “0” 的表示不唯一，不利于程序员编程
- 机器实现加、减运算的方法不统一
- 需对符号位进行单独处理，不利于硬件设计

❖ 反码

- 较少使用

❖ 补码

- “0” 的表示唯一
- 机器实现加、减运算的方法统一（模运算）
- 符号位参加运算，不需要单独处理

2.2 定点数

❖ 模运算

- 模：指一个计数系统所能表示的数据个数。有模运算是指运算结果超过模时，模（或模的整数倍）将溢出而只剩下余数。

8位二进制的模为 $2^8=256$

- 假设M为模，若数a, b满足 $a + b = M$ ，则称a, b互为补数。
- 在有模运算中，减去一个数等于加上这个数对模的补数。
 - 举例： $[127-2]_{\text{补}} = [127]_{\text{补}} + [-2]_{\text{补}}$

验证模运算： $[X \pm Y]_{\text{补}} = [X]_{\text{补}} + [\pm Y]_{\text{补}}$

其中X, Y为正负数均可，符号位参与运算
验证：

$$[X]_{\text{补}} = 2^n + X \pmod{2^n}$$

因此，

$$\begin{aligned} [X \pm Y]_{\text{补}} &= 2^n + (X \pm Y) \pmod{2^n} \\ &= (2^n + X) + (2^n \pm Y) \\ &= [X]_{\text{补}} + [\pm Y]_{\text{补}} \end{aligned}$$



- 时钟系统，模位12。
- 当前5点整，8小时后是1点，4小时前也是1点。
- $5+8=5-4=1 \pmod{12}$

2.3 浮点数表示

❖ **浮点数的一般表示法**：分为阶码和尾数两个部分

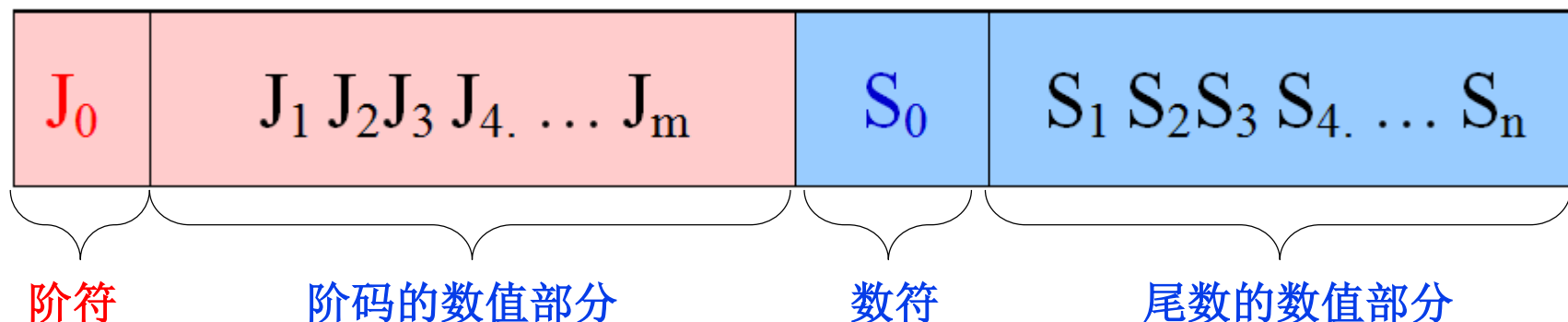
➤ 阶码：采用定点整数表示

➤ 尾数：采用定点小数表示

➤ 例： $(178.125)_{10} = (10110010.001)_2$
 $= 0.10110010001 \times 2^{01000}$

阶码：01000

尾数：0.10110010001



2.3 浮点数表示 (IEEE 754)

❖ IEEE 754: 符号 (Sign)、阶码 (Exponent) 和尾数 (Mantissa)。

单精度浮点数
32位

数符 S: 1位	阶 E: 8位	尾数m: 23位
-------------	---------	----------

双精度浮点数
64位

数符 S: 1位	阶 E: 11位	尾数m: 52位
-------------	----------	----------

❖ IEEE 754标准

- 数符 S: 1位, 0表示正数, 1表示负数
- 阶码 E: 移码表示, n 位阶码偏移量为 $2^{n-1}-1$ 。如8位阶码偏移量为 7FH (即127), 11位阶码偏移量3FFH (即1023)
- 尾数 M: 原码表示, 尾数必须规格化成小数点左侧一定为1, 并且小数点前面这个1作为隐含位被省略。这样单精度浮点数尾数实际上为24位。
- 规格化数 (尾数) 形式: $M=1.m$

2.3 浮点数表示 (IEEE 754标准)

❖ 浮点数精度

➤ 单精度浮点数表示公式: $(-1)^S \times 1.m \times 2^{(E-127)}$

➤ 双精度浮点数表示公式: $(-1)^S \times 1.m \times 2^{(E-1023)}$

❖ IEEE 754关于浮点数表示的约定 (单精度为例)

E	M	浮点数 N
$1 \leq E \leq 254$	$M \neq 0$	表示规范浮点数 $N = (-1)^S \times 1.m \times 2^{(E-127)}$
$E = 0$	$M = 0$	表示 $N = 0$
$E = 0$	$M \neq 0$	表示非规范浮点数 $N = (-1)^S \times 0.m \times 2^{-126}$
$E = 255$	$M = 0$	表示无穷大, 由符号位 S 确定是正无穷大还是负无穷大
$E = 255$	$M \neq 0$	NaN (Not a Number) 不是一个数

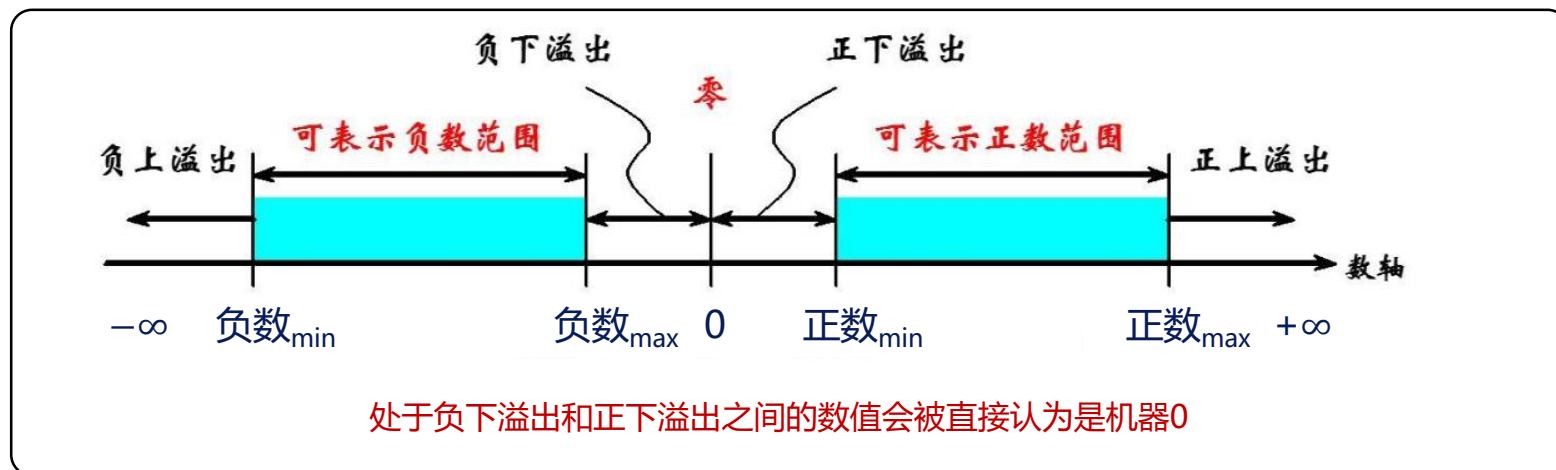
E的范围是0到255, 0和255两个值用于表示特殊值。

非规格化浮点数尾数不必小数点左侧为1, 而是0。

NaN 用于处理计算中出现的错误情况, 比如 0.0 除以 0.0 。

2.3 浮点数表示 (IEEE 754标准)

❖ 单精度浮点数表示范围



格式	负数 _{min}	负数 _{max}	正数 _{min}	正数 _{max}
单精度规格化 $(-1)^s \times 1.m \times 2^{E-127}$	$-(2 - 2^{-23}) \times 2^{254-127}$ $\approx -3.4 \times 10^{38}$ (S=1,E=254,m=11..11)	$-1.0 \times 2^{1-127}$ $= -2^{-126}$ (S=1,E=1,m=0)	$1.0 \times 2^{1-127}$ $= 2^{-126}$ (S=0,E=1,m=0)	$(2 - 2^{-23}) \times 2^{254-127}$ $\approx 3.4 \times 10^{38}$ (S=0,E=254,m=11..11)
单精度非规格化 $(-1)^s \times 0.m \times 2^{-126}$	$-(1 - 2^{-23}) \times 2^{-126}$ (S=1,E=0,m=11..11)	$-2^{-23} \times 2^{-126}$ $= -2^{-149}$ (S=1,E=0,m=00..01)	$2^{-23} \times 2^{-126}$ $= 2^{-149}$ (S=0,E=0,m=00..01)	$(1 - 2^{-23}) \times 2^{-126}$ (S=0,E=0,m=11..11)



1. 单精度规格化浮点数 负数_{max} 的表示是?
2. 单精度规格化浮点数 正数_{max} 的表示是?

2.3 浮点数表示 (IEEE 754标准)

❖ 单精度浮点数表示范围

符号 (S)	阶码 (E, 8位)	尾数 (m, 23位)	16进制表示	备注
0	00000000	00...00	00000000	正零
0	00000000	00...01~11...11	00000001~007FFFFFFF	非规格化正浮点数 $0.m \times 2^{-126}$
0	00000001~ 11111110	00...00~11...11	0080000~7F7FFFFFFF	规格化正浮点数 $1.m \times 2^{(E-127)}$
0	11111111	00...00	7F800000	正无穷大 (正上溢出)
0	11111111	00...01~11...11	7F800001~7FFFFFFFFF	+NaN
1	00000000	00...00	80000000	负零
1	00000000	00...01~11...11	80000001~807FFFFFFF	非规格化负浮点数 $-0.m \times 2^{-126}$
1	00000001~ 11111110	00...00~11...11	8080000~FF7FFFFFFF	规格化负浮点数 $-1.m \times 2^{(E-127)}$
1	11111111	00...00	FF800000	负无穷大 (负上溢出)
1	11111111	00...01~11...11	FF800001~FFFFFFFFFF	- NaN

2.3 浮点数表示 (IEEE 754标准)

❖ 单精度浮点数示例

$$(178.125)_{10} = (10110010.001)_2 \\ = 1.0110010001 \times 2^{111}$$

$$S = 0$$

$$E = 00000111 + 01111111 \\ = 10000110$$

$$m = 011001000100000000000000$$

0100 0011	0011 0010	0010 0000	0000 0000
-----------	-----------	-----------	-----------

十六进制: 43 32 20 00

$$(-0.0449219)_{10} = (-0.0000101110)_2 \\ = -1.01110 \times 2^{-101}$$

$$S = 1$$

$$E = -00000101 + 01111111 \\ = 01111010$$

$$m = 011100000000000000000000$$

1011 1101	0011 1000	0000 0000	0000 0000
-----------	-----------	-----------	-----------

十六进制: BD 38 00 00

2.4 非数值数据的表示

❖ 逻辑数据编码

- 一位二进制编码表示：真（1）、假（0）

❖ 西文字符编码（ASCII码，7位编码，128个字符）

- 数字字符：0~9（编码30H~39H）
- 英文字母：A~Z（编码41H~5AH），a~z（编码61H~7AH）
- 专用符号：+/-/%/*/&/...
- 控制字符（不可打印或显示字符）

❖ 汉字编码

- 输入码：用于汉字的输入，如拼音码、五笔字型码；
- 国标码：1981年我国颁布了《信息交换用汉字编码字符集·基本集》（GB2312—80）。该标准规定了6763个常用汉字的标准代码。
- 内 码：用于汉字存储、查找、传送等，基于国标码，占2个字节；
- 点阵码或汉字向量描述：用于汉字的显示和打印。

❖ 数据信息的校验码

- 奇偶校验码：简单的奇偶校验可以检测出任意奇数位错误，不能检测出偶数位错误，不能纠错
- 海明校验码
- 循环冗余校验码

第一部分：概述

一. 计算机组成与结构简介

1. 计算机的基本组成
2. 计算机的层次结构

二. 计算机中数据信息的表示

1. 无符号数和有符号数
2. 定点数、浮点数表示
3. 非数值数据的表示

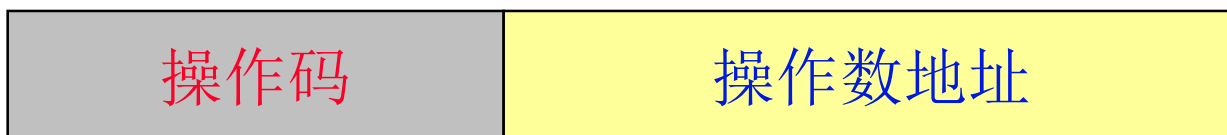
三. 计算机的基本工作过程

1. 指令的含义
2. 程序的执行

3.1 计算机的工作过程

❖ **机器指令**：计算机硬件可以执行的基本操作的二进制代码。

- 指令格式：操作码 + 操作数（操作数地址）
- 操作码：指明指令的操作性质
- 操作数（地址）：指令操作数的位置（或操作数本身）

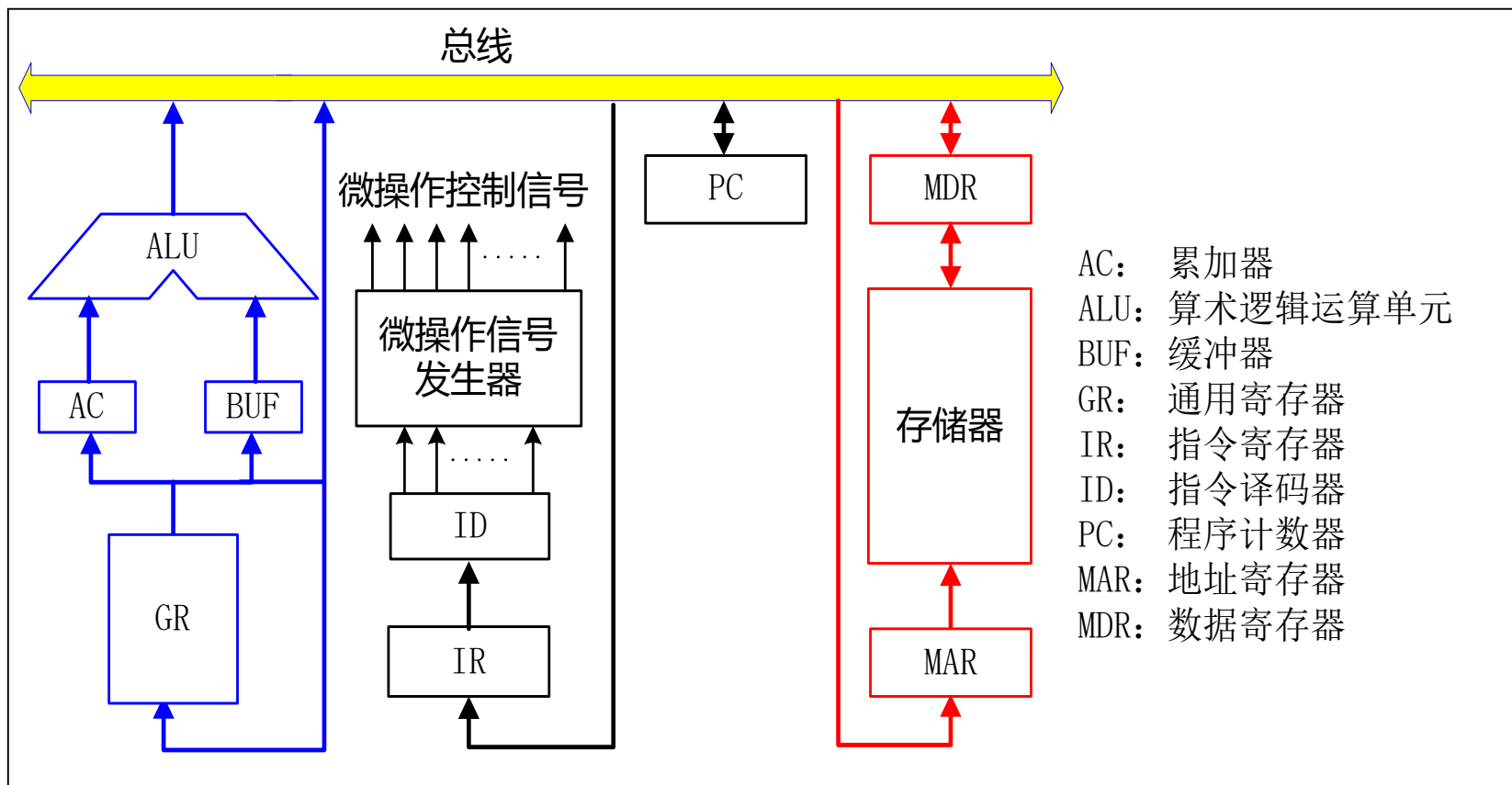


❖ **程序**：在此特指一段机器指令序列。

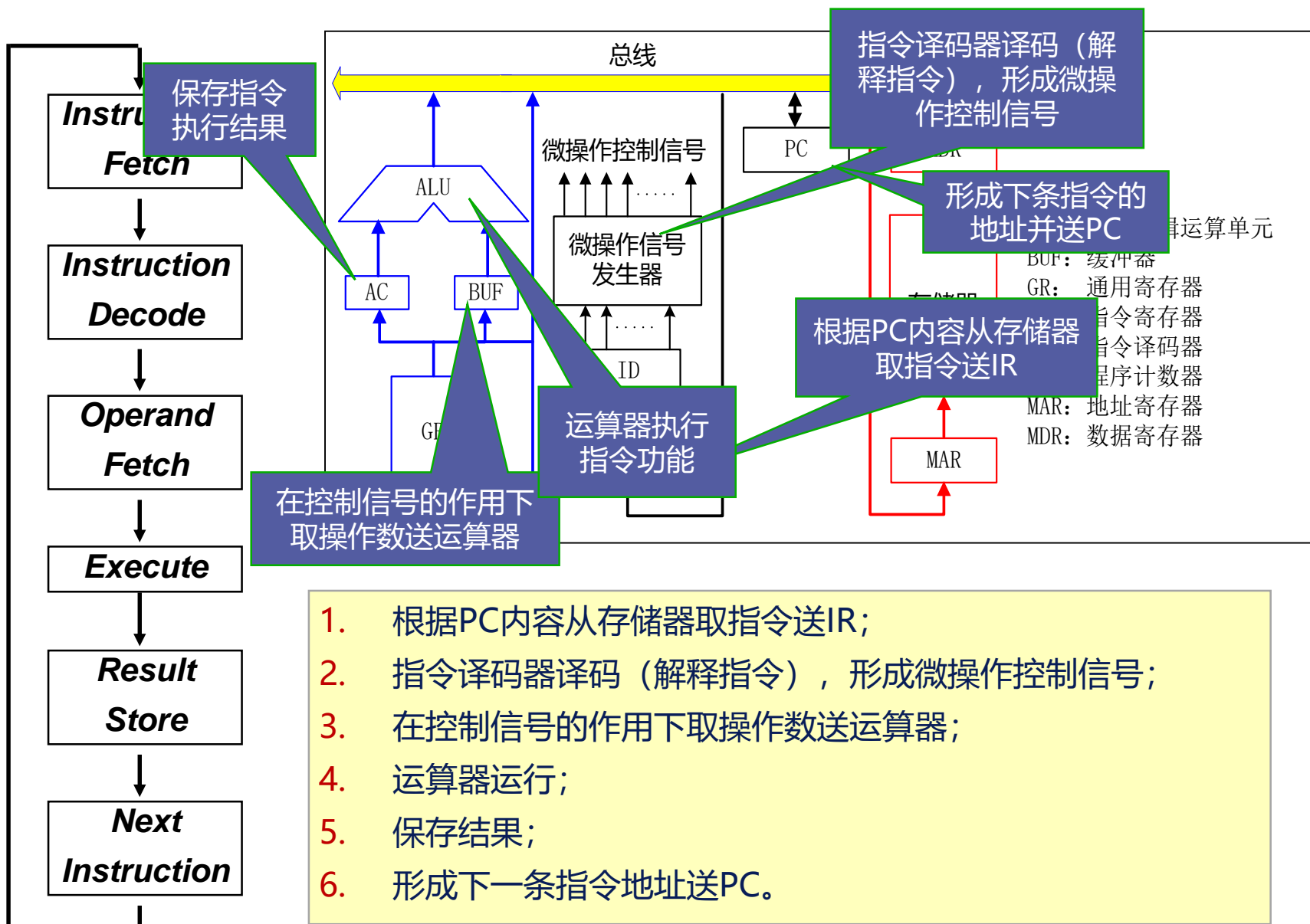
- 完成一定的功能，采用某种算法，具备一定的流程；
- 计算机按照程序所规定的流程和指令顺序，一条一条地执行指令，达到完成程序所规定的功能的目的。
- 计算机采用程序计数器（Program Counter）来决定指令执行的顺序。

3.2 指令的执行过程

机器结构简化图



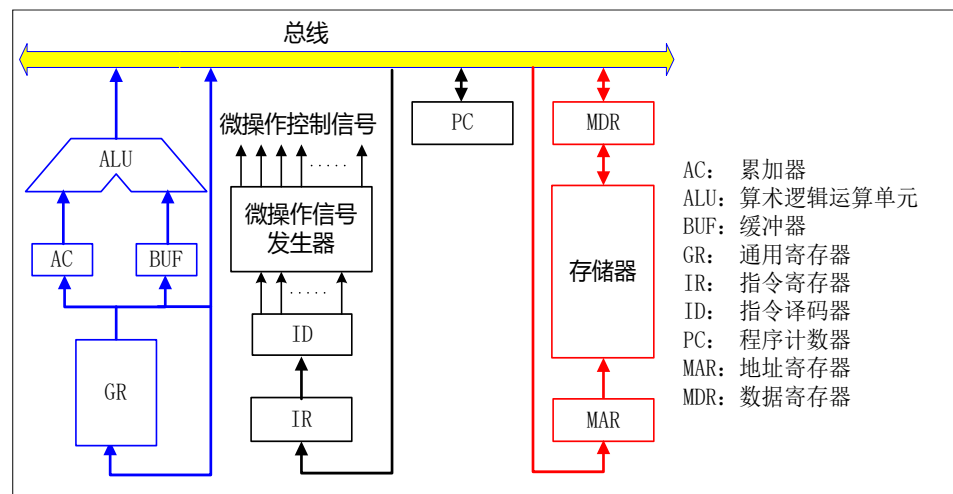
3.2 指令的执行过程



Example

某计算机字长16位，机器结构和指令格式如图，计算 $Y=ax^2+bx-c$

- 假定： $a=15(000FH)$ ， $b=96(0060H)$ ， $c=72(0048H)$ ， $x=8(0008H)$ ，已存放在地址为12H开始的连续存储单元中；
- 结果y的存储单元地址为10H。



指令系统：16位指令系统

操作码	操作数地址
-----	-------

8位

8位

指令	操作码	说明
ADD	00H	$AC \leftarrow (AC) + Mem(Add)$
LD	01H	$AC \leftarrow Mem(Add)$
SUB	02H	$AC \leftarrow (AC) - Mem(Add)$
MUL	03H	$AC \leftarrow (AC) \times Mem(Add)$
ST	04H	$Mem(Add) \leftarrow (AC)$

内存

单元地址

	00H
	02H
	04H
	06H
	08H
	0AH
	0CH
	0EH
结果y将存放在此	10H
000FH (a)	12H
0060H (b)	14H
0048H (c)	16H
0008H (x)	18H

Example

某计算机字长16位，机器结构和指令格式如图，计算 $Y=ax^2+bx-c$

- 假定： $a=15(000FH)$ ， $b=96(0060H)$ ， $c=72(0048H)$ ， $x=8(0008H)$ ，已存放在地址为12H开始的连续存储单元中；
- 结果y的存储单元地址为10H。

指令	操作码	说明
ADD	00H	$AC \leftarrow (AC) + \text{Mem}(\text{Add})$
LD	01H	$AC \leftarrow \text{Mem}(\text{Add})$
SUB	02H	$AC \leftarrow (AC) - \text{Mem}(\text{Add})$
MUL	03H	$AC \leftarrow (AC) \times \text{Mem}(\text{Add})$
ST	04H	$\text{Mem}(\text{Add}) \leftarrow (AC)$

操作步骤	指令	代码
$AC \leftarrow a$ 获取a	LD a	0112H
$AC \leftarrow (AC) \times x$ 计算得到ax	MUL x	0318H
$AC \leftarrow (AC) + b$ 计算得到ax+b	ADD b	0014H
$AC \leftarrow (AC) \times x$ 计算得到 ax^2+bx	MUL x	0318H
$AC \leftarrow (AC) - c$ 计算得到 ax^2+bx-c	SUB c	0216H
$\text{Mem} \leftarrow (AC)$ 保存结果	ST y	0410H

分配程序
存储地址

00H

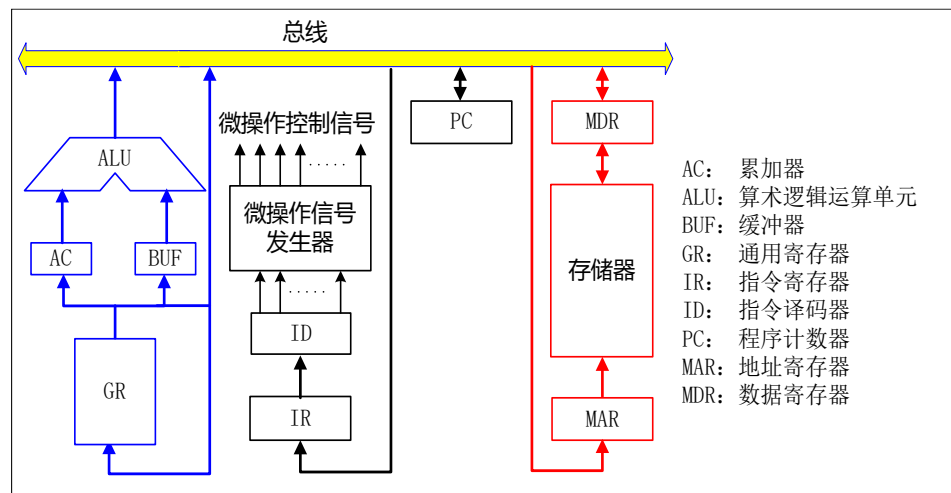
02H

04H

06H

08H

0AH



AC: 累加器
ALU: 算术逻辑运算单元
BUF: 缓冲器
GR: 通用寄存器
IR: 指令寄存器
ID: 指令译码器
PC: 程序计数器
MAR: 地址寄存器
MDR: 数据寄存器

内存	单元地址
0112H	00H
0318H	02H
0014H	04H
0318H	06H
0216H	08H
0410H	0AH
	0CH
	0EH
结果y将存放在此	10H
000FH (a)	12H
0060H (b)	14H
0048H (c)	16H
0008H (x)	18H

Example

操作步骤	指令	代码
$AC \leftarrow a$ 获取a	LD a	0112H
$AC \leftarrow (AC) \times x$ 计算得到ax	MUL x	0318H
$AC \leftarrow (AC) + b$ 计算得到ax+b	ADD b	0014H
$AC \leftarrow (AC) \times x$ 计算得到ax ² +bx	MUL x	0318H
$AC \leftarrow (AC) - c$ 计算得到ax ² +bx-c	SUB c	0216H
$Mem \leftarrow (AC)$ 保存结果	ST y	0410H

开始PC

PC

PC

PC

PC

PC

PC

累加器AC

$AC \leftarrow ax^2 + bx - c$ 0678H

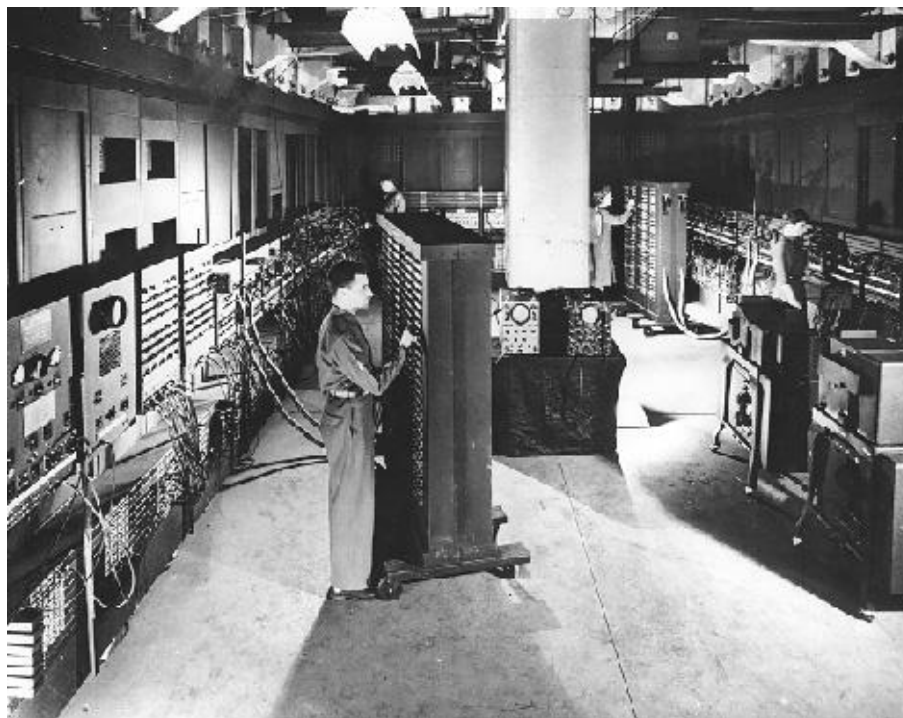
内存

地址

00H	→	0112H	00H
02H	→	0318H	02H
04H	→	0014H	04H
06H	→	0318H	06H
08H	→	0216H	08H
0AH	→	0410H	0AH
0CH	→		0CH
			0EH
		0678H (y)	10H
		000FH (a)	12H
		0060H (b)	14H
		0048H (c)	16H
		0008H (x)	18H

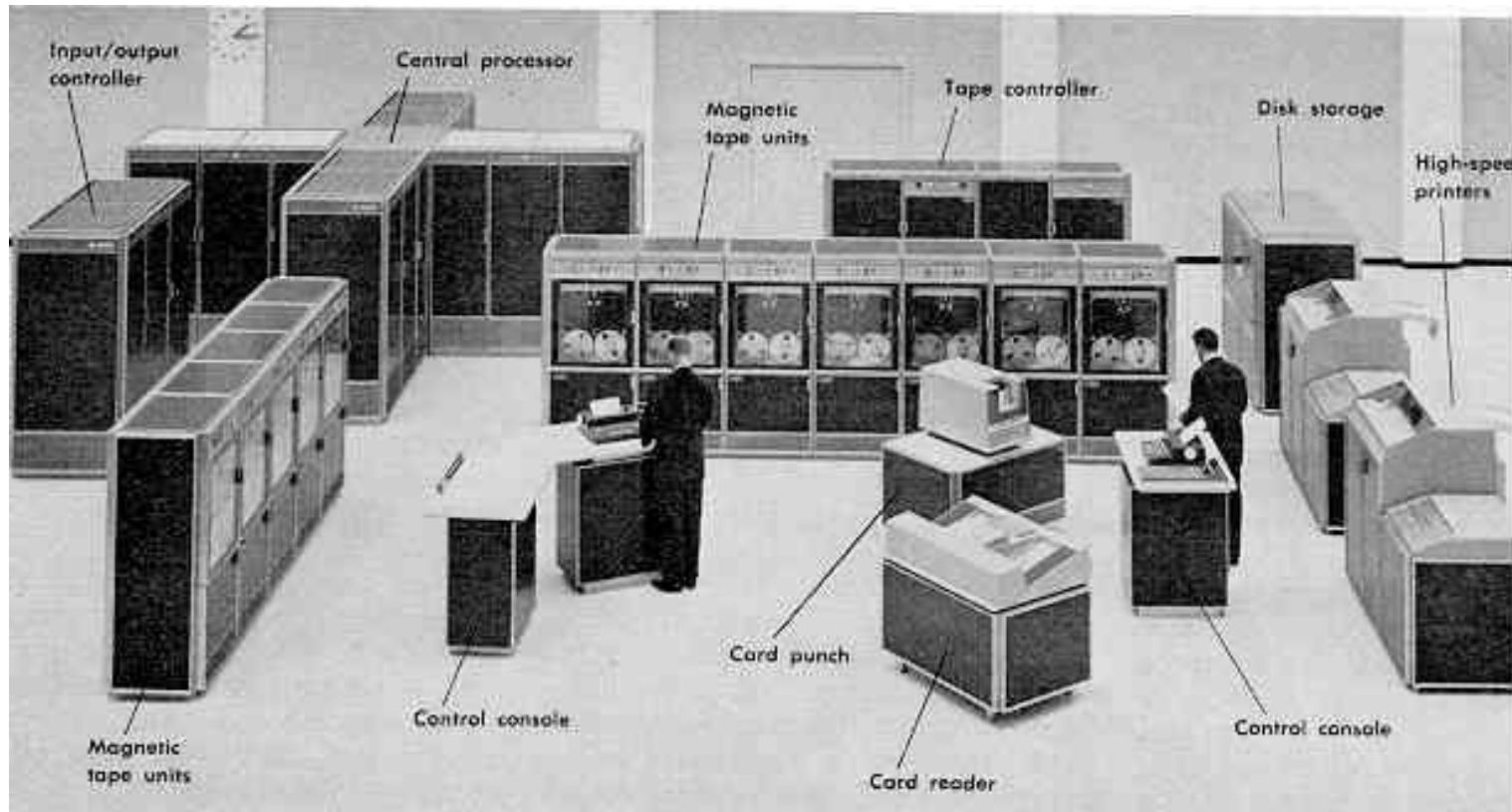
ENIAC: 第一台通用电子计算机

- ENIAC: 1946年, 世界上第一台通用电子计算机也即电子数字积分计算机。十进制 (非二进制) 计算机, 用十个电子管 (一个ON, 其余OFF) 表示一位十进制数, 算术运算按十进制的方式完成。
- 占地170平方米, 重30吨, 耗电140千瓦, 每秒可进行5000次加减法运算。共用18800个电子管, 70000个电阻, 10000个电容, 1500个继电器, 6000个开关, 50万条连线。



ENIAC: Electronic Numerical Integrator And Computer

Mainframe Era: 1950s - 1960s



Enabling Tech: Computers

Big Players: “Big Iron” (IBM, UNIVAC)

Cost: \$1M, Target: Businesses

Using: COBOL, Fortran, timesharing OS

Minicomputer Era: 1970s



Enabling Tech: Integrated circuits

Big Players: Digital, HP

Cost: \$10k, Target: Labs & universities

Using: C, UNIX OS

PC Era: Mid 1980s - Mid 2000s



Enabling Tech: Microprocessors

Big Players: Apple, IBM

Cost: \$1k, Target: Consumers (1/person)

Using: Basic, Java, Windows OS

Moore's Law

1965年，摩尔预言半导体芯片上集成的晶体管数量将每年增加一倍。经过十年的观察，1975年摩尔将每年修正为每两年，业界将这个现象定义成摩尔定律。（后来业界用平均值18个月）



Gordon E. Moore

The experts look at

Cramming onto inte

With unit cost falling
circuit rises, by 1975
many as 65,000 com

By Gordon E. Moore
Director, Research and Development
division of Fairchild Camera and

The future of integrated electronics lies in the proliferation of electronics in new areas.

Integrated circuits will be computers—or at least terminators—automatic controls for portable communications equipment needs only a display to be the biggest potential systems. In telephone communications, integrated circuits will perform data processing.

Computers will be more important in completely different ways of integrated electronics may

The author

Dr. Gordon E. Moore is one of the new breed of electronic engineers, schooled in the physical sciences rather than electronics. He earned a B.S. degree in chemistry from the University of California and a Ph.D. degree in physical chemistry from the California Institute of Technology. He was one of the founders of Fairchild Semiconductor and has been director of the research and development laboratories since 1959.

The establishment

Integrated electronics is established today. Its techniques are almost mandatory for new military systems, since their reliability, size and weight required by some of them is achievable only with integration. Such programs as Apollo, for manned moon flight, have demonstrated the reliability of integrated electronics by showing that complete circuit functions are as free from failure as the best individual transistors.

Most companies in the commercial computer field have machines in design or in early production employing integrated electronics. These machines cost less and perform better than those which use "conventional" electronics.

Instruments of various sorts, especially the rapidly increasing number employing digital techniques, are starting to use integration because it cuts costs of both manufacture and design.

The use of linear integrated circuitry is still restricted primarily to the military. Such integrated functions are expensive and not available in the variety required to satisfy a major fraction of linear electronics. But the first applications are beginning to appear in commercial electronics, particularly in equipment which needs low-frequency amplifiers of small size.

Reliability counts

In almost every case, integrated electronics has demonstrated high reliability. Even at the present level of production—low compared to that of discrete components—it often reduces systems cost, and in many systems improved performance has been realized.

Integrated electronics will make electronic techniques more generally available throughout all of society, performing many functions that presently are done inadequately by other techniques or not done at all. The principal advantages will be lower costs and greatly simplified design—payoffs from a ready supply of low-cost functional packages.

For most applications, semiconductor integrated circuits will predominate. Semiconductor devices are the only reasonable candidates presently in existence for the active elements of integrated circuits. Passive semiconductor elements look attractive too, because of their potential for low cost and high reliability, but they can be used only if precision is not a prime requisite.

Silicon is likely to remain the basic material, although others will be of use in specific applications. For example, gallium arsenide will be important in integrated microwave functions. But silicon will predominate at lower frequencies because of the technology which has already evolved around it and its oxide, and because it is an abundant and relatively inexpensive starting material.

Costs and curves

Reduced cost is one of the big attractions of integrated electronics, and the cost advantage continues to increase as the technology evolves toward the production of larger and larger circuit functions on a single semiconductor substrate. For simple circuits, the cost per component is nearly inversely proportional to the number of components, the result of the

equivalent piece of semiconductor in the equivalent package containing more components. But as components are added, decreased yields more than compensate for the increased complexity, tending to raise the cost per component. Thus there is a minimum cost at any given time in the evolution of the technology. At present, it is reached when 50 components are used per circuit. But the minimum is rising rapidly while the entire cost curve is falling (see graph below). If we look ahead five years, a plot of costs suggests that the minimum cost per component might be expected in circuits with about 1,000 components per circuit (providing such circuit functions can be produced in moderate quantities). In 1970, the manufacturing cost per component can be expected to be

only

cross

graph

can be

longer

done

cost

number

will be

the

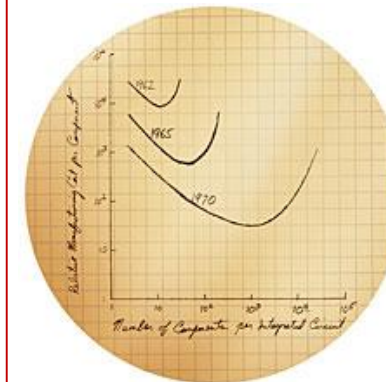
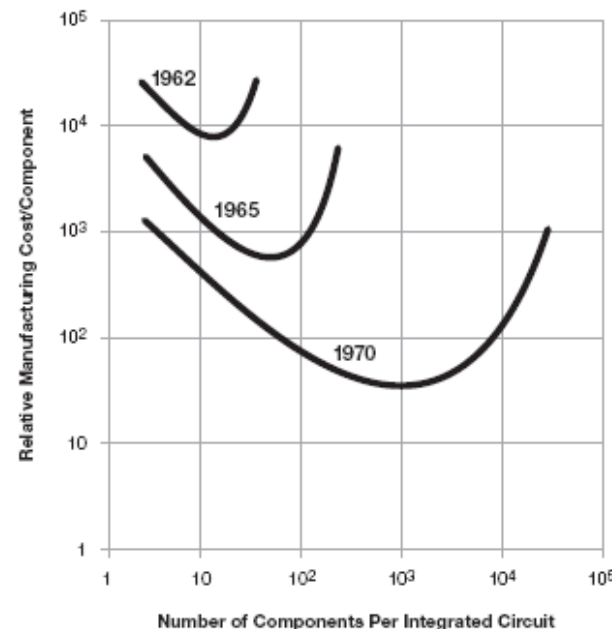
value

two-

very

in its

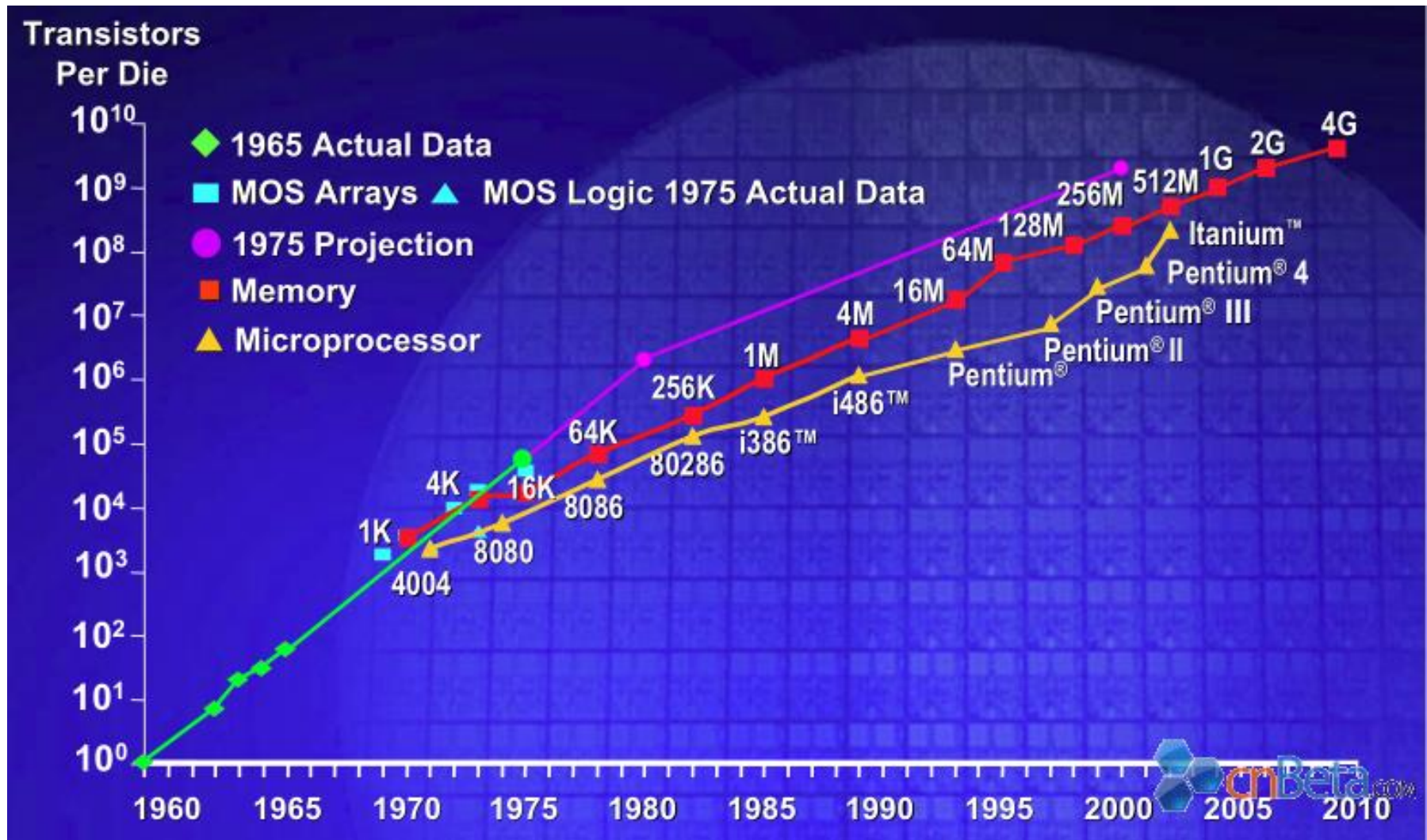
can be



Electronics, Volume 38, Number 8

Electronics, Volume 38, Number 8, April 19, 1965

Moore's Law



中国计算机发展的历史记忆

➤ 第一代电子管计算机(1958-1964年)

- 1958年8月：103型计算机(即DJS-1型)，中国科学院计算所和北京有线电厂（原738厂）共同研制，字长31位，内存容量1024字节，运算速度每秒3000次。**曲阜师范大学中国教师博物馆保存一台完整的103机。**
- 1959年10月：104型计算机，共4200个电子管，4000个晶体二极管，由22个机柜组成，字长39位，每秒运行1万次。



➤ 第二代晶体管计算机(1965-1972年)

- 1965年：我国第一台大型晶体管计算机(109乙机)成功研制，共用2万多支晶体管，3万多支二极管。



➤ 第三代中小规模集成电路计算机(1973-80年代初)

- 1975年：DJS-131小型机（DJS-100系列小型机的主要型号），字长16位、内存4~32K字（可扩至64K字）、运算速度每秒50万次。DJS 100系统开创了我国计算机工业化系列化设计与生产的先河，该系列计算机在我国计算机发展史具有非常重要的历史意义。**北航虚拟现实技术与系统国家重点实验室计算机博物馆保存一台完整的DJS-131小型机。**
- 1983年：“银河-1”亿次计算机，1978年由邓小平亲自批准研制，由国防科技大学负责，全国20多家单位参与，1983年研制成功，是我国第一台自主研制的亿次计算机系统。**国防科技大学计算机学院院史馆保存一台完整的“银河-1”亿次机。**



➤ 第四代超大规模集成电路计算机(80年代中期至今)

- 微型计算机：1984年，长城0520CH微型计算机（兼容IBM PC）发布，后续长城286，长城386，长城486、联想386，联想486等兼容微型计算机陆续研制发布。
- 高性能计算机：1992年国防科大研究成功银河-II通用并行巨型机，峰值速度达每秒4亿次浮点运算，1997年银河-III百亿次并行巨型计算机，峰值性能为每秒130亿次浮点运算；2009年，中国首台千万亿次计算机“天河一号”研制成功；
- 曙光系统超级服务器、“神威”大规模并行计算机系统。



课程教学内容视图

