

复习

0 了解计算机的简史

1 数据存储

1.1 位和位存储

- 布尔运算：与、或、非、异或
- 门：门的组合可以实现怎样逻辑？
- 存储单元：位、字节、地址
- 数制：二进制、八进制、十六进制、十进制
- 存储器的容量：K、M、G

例：拥有4KB的计算机存储器里有多少个二进制位？

1.2 用位模式表示信息

- 文本：ASCII —7位（扩展8位，最高位为0）即1个字节
Unicode—16位，即2个字节
- 数值表示
- 二进制系统：数制转换
- 整数存储：原码、补码、溢出
- 小数存储：定点小数、浮点小数

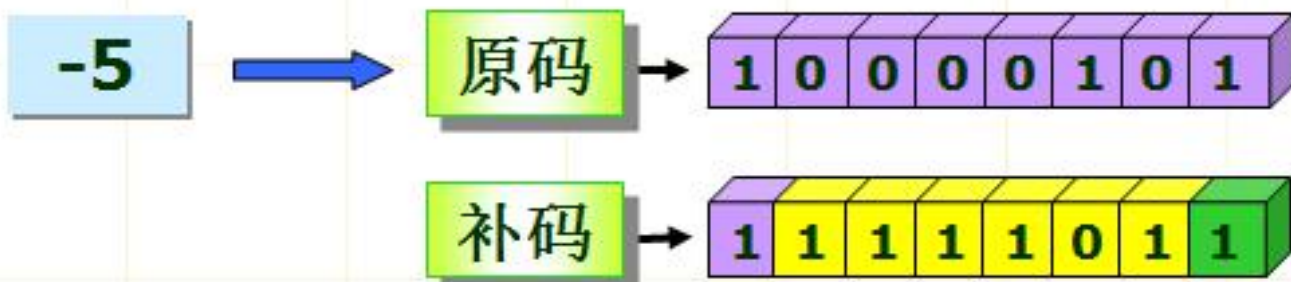
原码和补码

为了运算方便，计算机中机器数采用补码表示。

原码：正号为0，负号为1，数值部分为二进制绝对值。

补码：正数的补码和原码相同；负数的补码是将其原码除符号位外各位取反，末位加1。

+5的原码和补码都是00000101



练习题

- 1、二进制**101.111**对应的十进制数为 **5.875**。
- 2、十进制**65**对应的二进制数为 **1000001**，
假如用**ASCII**码表示，编码为 **00110110 00110101**。
- 3、十进制数**-17**二进制补码为 **11101111**。
- 4、执行如下补码加法运算，观察是否溢出

00111

+01100

10011

溢出

10111

+11010

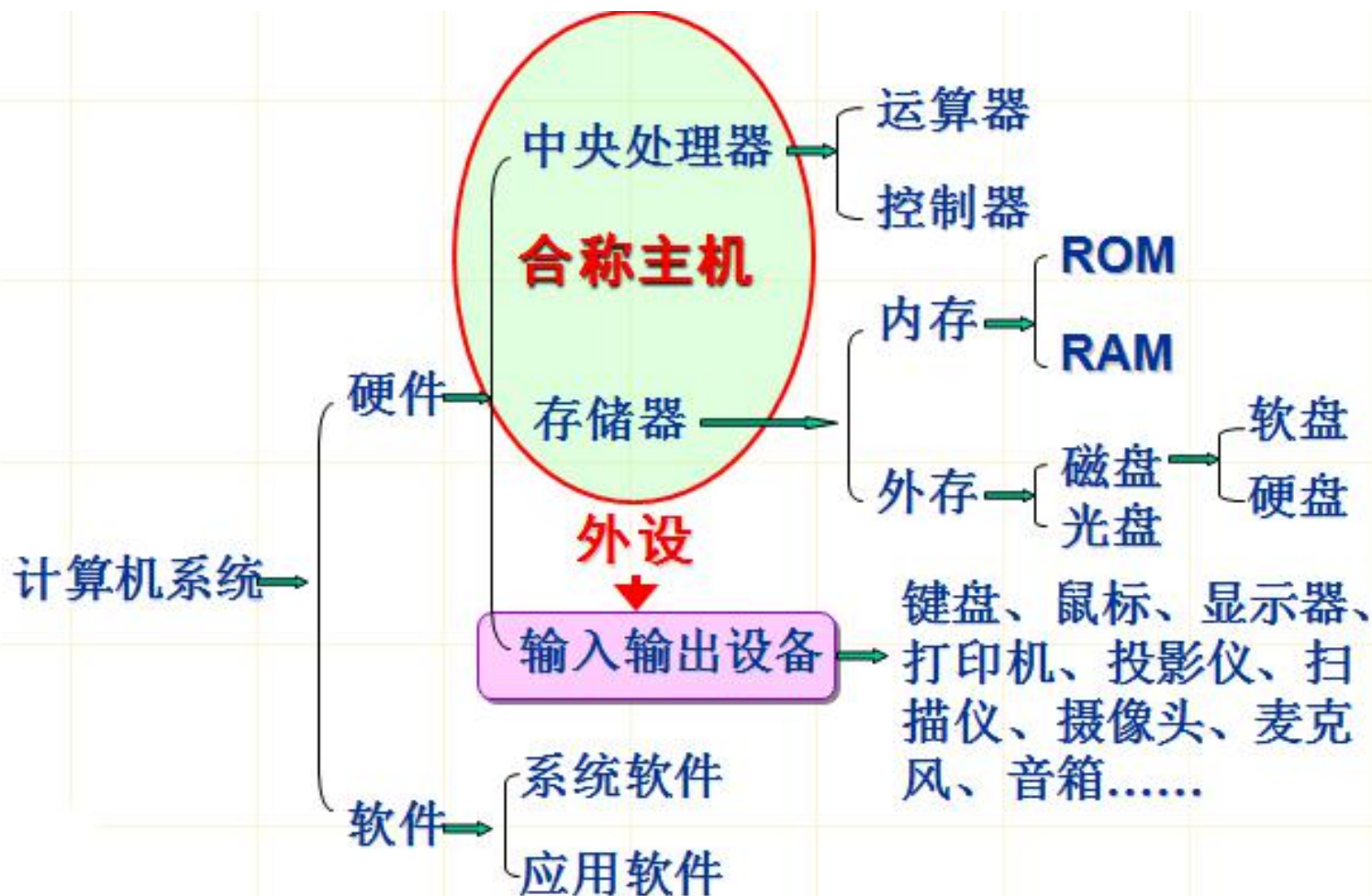
100001

溢出

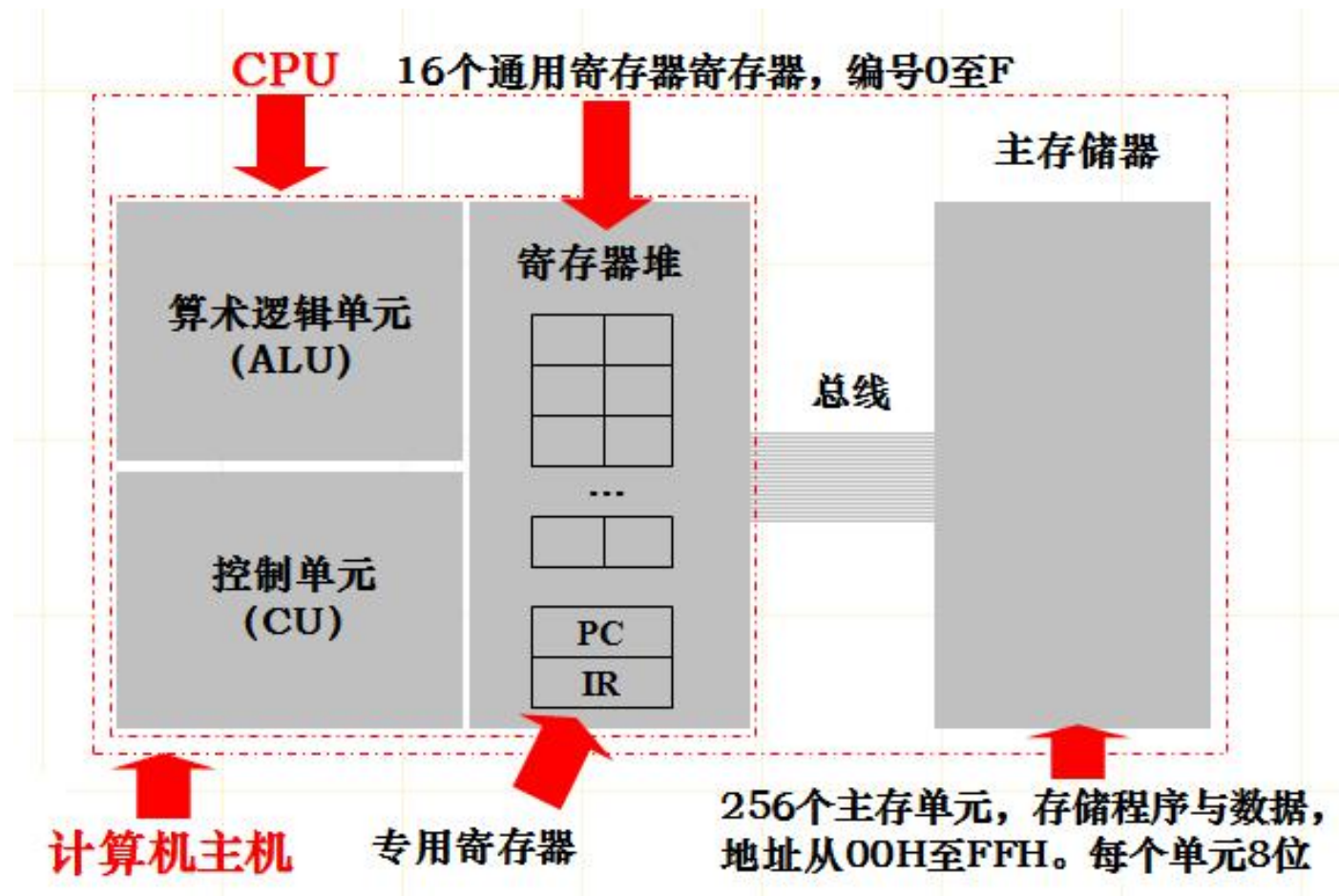
注意：当相同符号两值运算结果符号位变化，则溢出。

2 数据操控

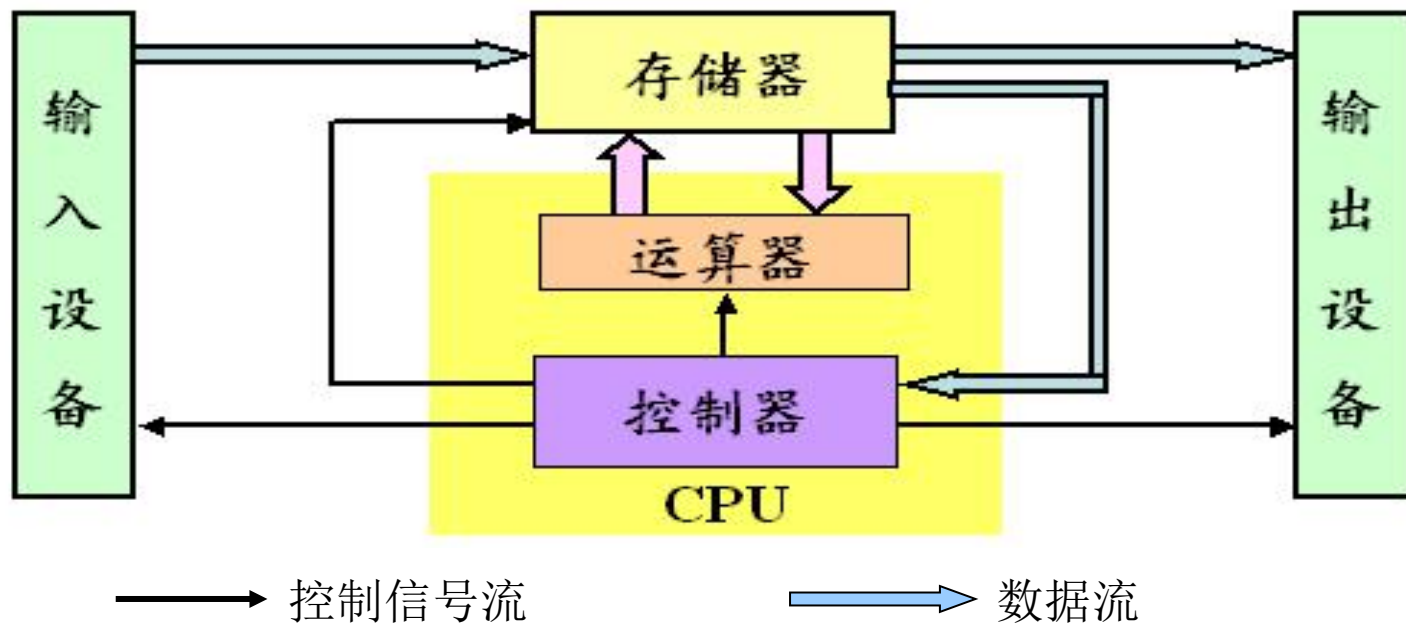
2.1 计算机体系结构



- CPU:



- 冯诺依曼结构



计算机由运算器、控制器、存储器、输入设备、输出设备五大部分组成

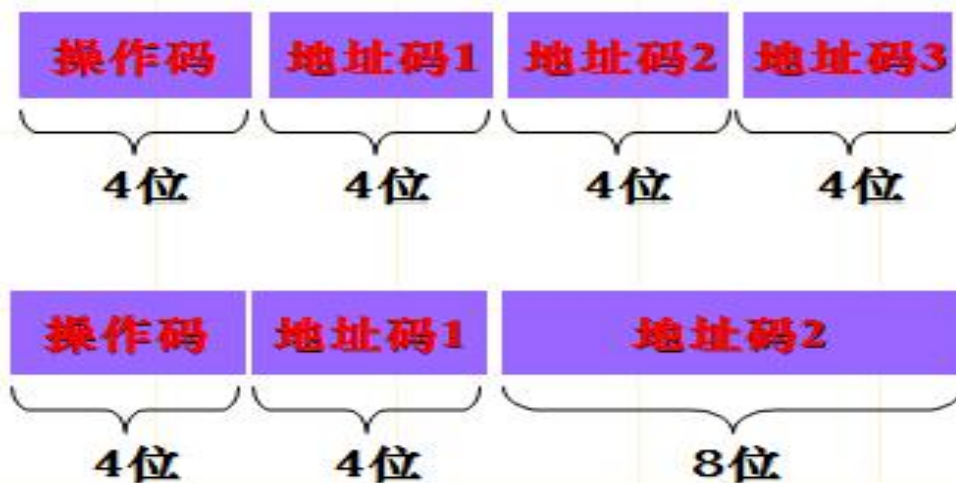
2 数据操控

2.2 机器语言

- 指令系统

机器指令格式——各字段位数

以附录C为例：



指令系统---以附录C为例

0001	R	XY
-------------	----------	-----------

将地址为XY的主存单元数据读出并送至寄存器R

0010	R	XY
-------------	----------	-----------

将操作数XY送至寄存器R

0011	R	XY
-------------	----------	-----------

将寄存器R中的值存至主存地址为XY的单元

0100		S	T
-------------	--	----------	----------

将寄存器S中的值复制到寄存器T中

0101	R	S	T
-------------	----------	----------	----------

将寄存器S的值与寄存器T的值进行补码加，结果存至寄存器R中

0110	R	S	T
-------------	----------	----------	----------

将寄存器S的值与寄存器T的值作为浮点数进行相加，结果存至寄存器R中

.....

0000	0000 0000 0000
-------------	-----------------------

程序停止执行

操作序列如下：

- 1、将主存6C单元数据3（二进制0000 0011）**读出**，通过总线，送3号寄存器（临时存放，也可以是其它寄存器）；
- 2、将主存6D单元数据5（二进制0000 0101）**读出**，通过总线，送4号寄存器（临时存放，也可以是其它寄存器）；
- 3、启动ALU的加法电路，3号和4号寄存器的存数作为输入，结果存至0号寄存器；
- 4、将结果从0号寄存器读出，**写入**主存6E单元；
- 5、停止。

程序代码

<u>0001</u>	<u>0011</u>	<u>0110</u>	<u>1100</u>
1	3	6	C

<u>0001</u>	<u>0100</u>	<u>0110</u>	<u>1101</u>
1	4	6	D

<u>0101</u>	<u>0000</u>	<u>0011</u>	<u>0100</u>
5	0	3	4

<u>0011</u>	<u>0000</u>	<u>0110</u>	<u>1110</u>
3	0	6	E

<u>1100</u>	<u>0000</u>	<u>0000</u>	<u>0000</u>
C	0	0	0

• 执行过程

程序的执行过程

每条指令占两个存储单元，
每次取两个单元内容。

1、程序计数器 (PC) 存放第一条指令的主存地址(00H)。



2、取出00H单元中的指令，送至指令寄存器。



3、指令译码，同时程序计数器+2。



4、执行译码后的指令，将主存6CH单元内容 (3) 取到寄存器R3中。



4、取出02H单元中的指令，送指令寄存器。



5、指令译码，同时程序计数器+2。



6、执行译码后的指令，将主存6DH单元的内容（5）取到寄存器R4中。



8、取出主存04H单元中的指令，放到指令寄存器。



9、指令译码，同时程序计数器+2，指向主存06H单元。



10、执行译码后的指令，将3、4号寄存器内容送加法器相加，结果放到寄存器R0中。



11、取出主存06H单元中的指令，将它放到指令寄存器。



12、指令译码，同时，程序计数器+2，指向主存08H单元。



12、执行译码后的指令，将R0的内容（8）放到地址为6E的存储单元中。



14、取出主存08H单元中的指令，放到指令寄存器中。



15、指令译码，同时程序计数器+2，指向0AH单元。



16、执行译码后的指令，停机，程序完成。



2 数据操控

2.3 程序执行

1、将主存10H单元指令 (B320H) 取至指令寄存器IR中。



2、指令译码，同时程序计数器+2。



2、执行译码后的指令。由于R3与R0内容相同，因此，将指令寄存器中地址码20H送至程序计数器中，PC值为20H。下次执行主存20H单元中的指令。



说明：

指令B320H（助记符JMP R3 20H）为**条件转移指令**，即是否转移取决于条件是否成立（R3=R0?）。除此之外，在一般机器的指令系统中，还有**无条件转移指令**，如JMP 20H（助记符指令），即指令执行时，不作任何判断，直接将20H存至PC中，下次执行20H单元里的指令。

2 数据操控

2.4 运算：移位

◆ 循环移位 (**circular shift**)

- 将右（左）侧移出的位放置在左（右）端的空位上；

◆ 逻辑移位 (**logical shift**)

- 丢弃移出边界的位，空位用**0**填充；

◆ 算术移位 (**arithmetic shift**)

- 同逻辑移位，但是保留符号位不变。

3 操作系统

- 概念：定义、功能
- 发展：多道程序设计技术、分时技术
- 系统启动过程
- 进程状态转换

4 网络

- 计算机网络
- 网络拓扑：总线、环型、星型
- 什么是协议？
- 进程间的通信：客户机/服务器，对等
- IP地址
- DNS—域名系统
- Internet应用：E-mail、FTP、WWW等
- 超文本、浏览器、服务器、URL (统一资源定位)
- HTML (超文本标记语言)
- OSI和TCP/IP