

# 计算机组成原理与系统结构

## 第四章

## 运算方法与运算器

<http://jpkc.hdu.edu.cn/computer/zcyl/dzkjdx/>





## 第 4 章 运算方法与运算器

4. 1

定点数的加减运算及实现

4.

定点数的乘法运算及实现

4. 3

定点数除法运算及  
实现

4. 4

定点运算器的组成与结构

4.

浮点运算及运算器

4.

浮点运算器举例

本章小结

BACK



## 4.4 定点运算器的组成与结构



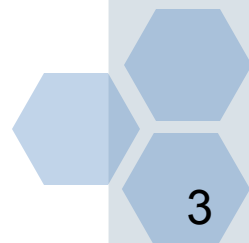
定点运算器的组成



定点运算器的内部总线结构与通路



标志寄存器

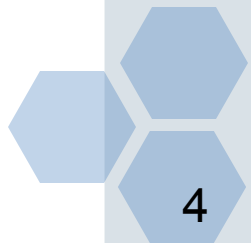




# 一、定点运算器的组成

## ❖ 基本组成包括：

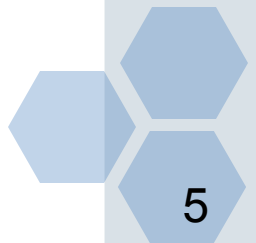
- **算术逻辑运算单元 ALU**：核心部件，实现算术运算和逻辑运算
- **暂存器**：用来存放参与计算的数据及运算结果，它只对硬件设计者可见，即只被控制器硬件逻辑控制或微程序所访问
- **通用寄存器堆**：用于存放程序中用到的数据，它可以被软件设计者所访问。
- **内部总线**：用于连接各个部件的信息通道。
- 其他可选电路





# 一、定点运算器的组成

- ❖ 设计定点运算器，如何确定各部件的功能和组织方式是关键，这取决于以下几个方面：
  - 指令系统
  - 机器字长
  - 机器数及其运算原理
  - 体系结构





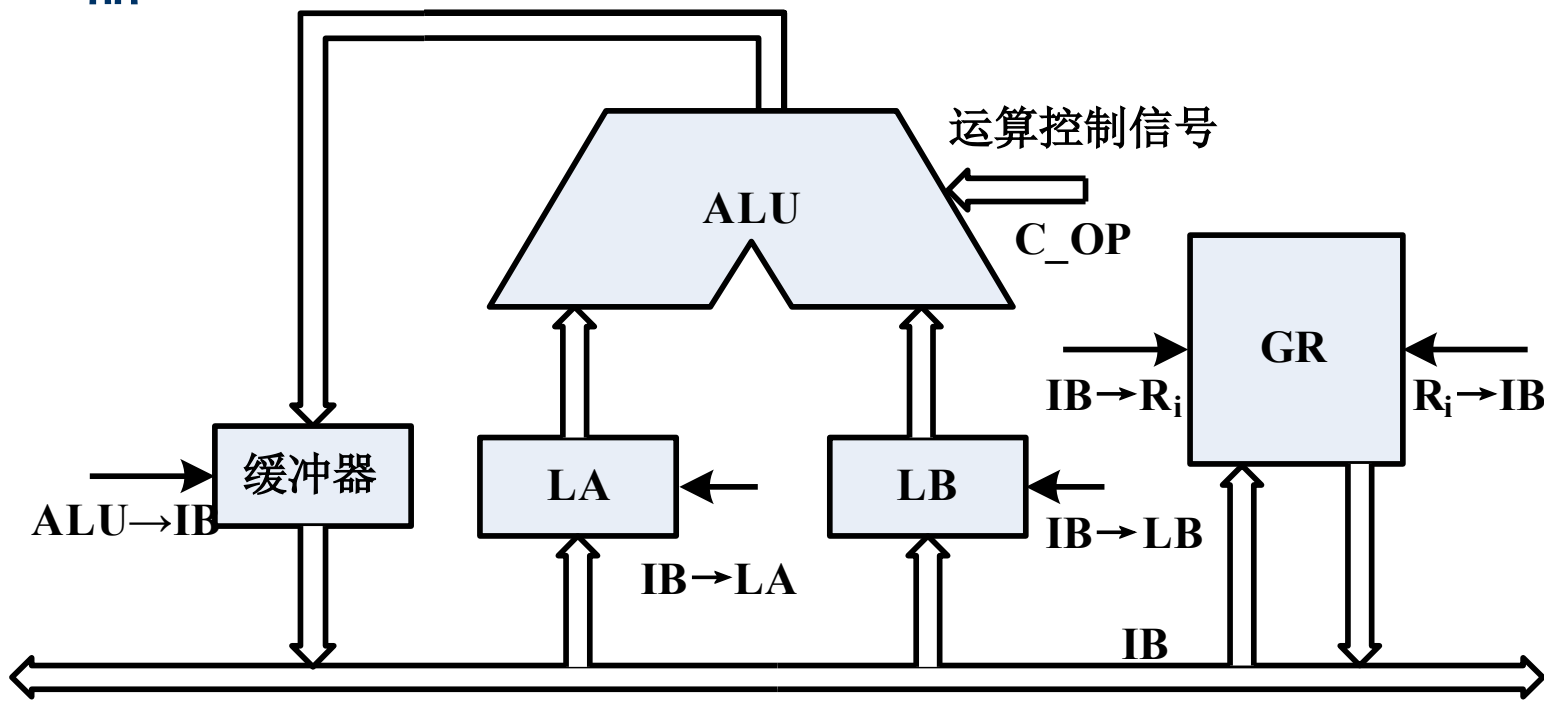
## 二、定点运算器的总线结构

### ❖ 1、单总线结构

- 单总线运算器的结构形式 1：ALU+2 个暂存器

❖ 如何完成运算操作：

$(R_i) \theta (R_j) \rightarrow R_k$   
?

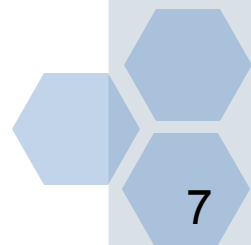




## 单总线运算器的结构形式 1— 数据通路

✧  $(R_i) \theta (R_j) \rightarrow R_k$  : 需要 3 步

序号	操作	发送的信号
1	$(R_i) \rightarrow LA$ ;	$R_i \rightarrow IB, IB \rightarrow LA$
2	$(R_j) \rightarrow LB$ ;	$R_j \rightarrow IB, IB \rightarrow LB$
3	ALU 运算, 结果 $\rightarrow R_k$	$C\_OP(\theta), ALU \rightarrow IB$ , $IB \rightarrow R_i$

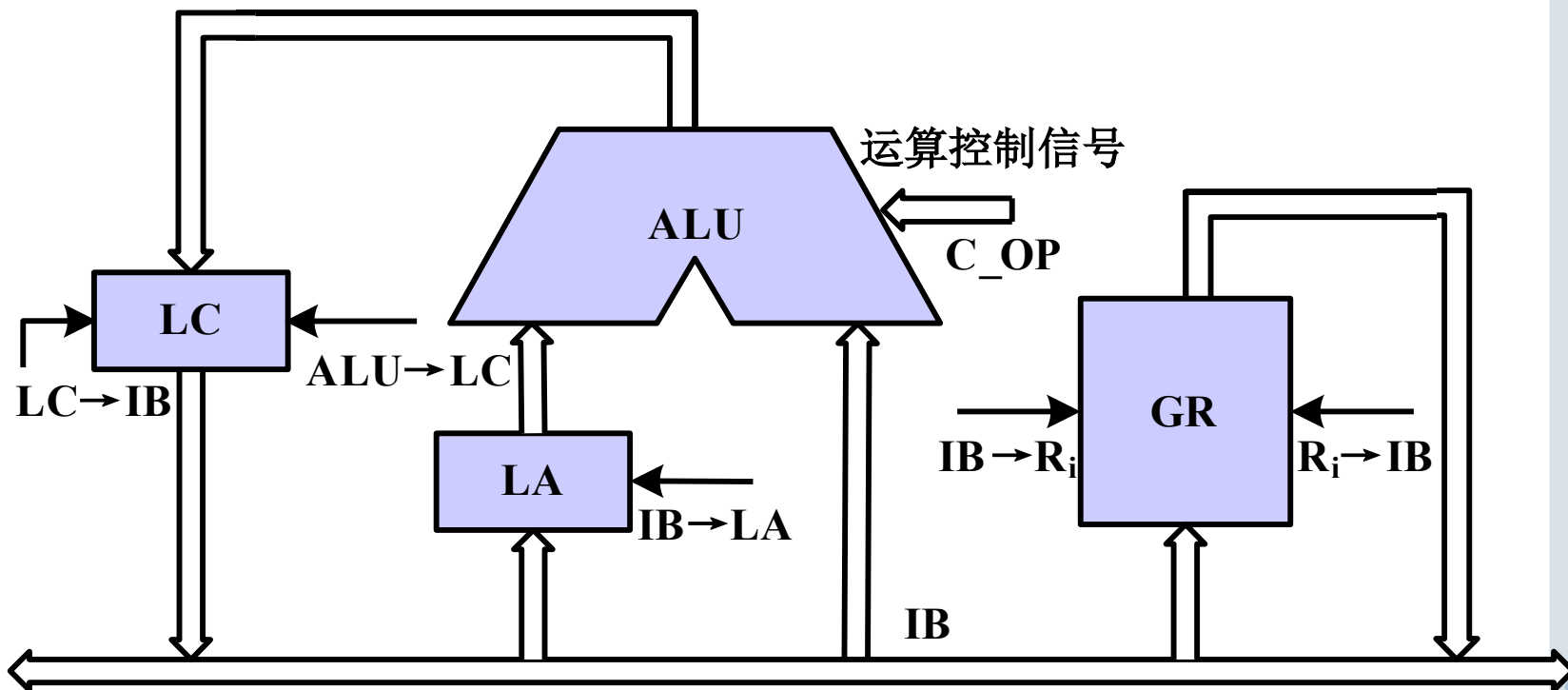




## 二、定点运算器的总线结构

- 单总线运算器的结构  
形式 2 : ALU+2 个暂  
存器

❖ 如何完成运算操作：  
 $(R_i) \theta (R_j) \rightarrow R_k$   
?



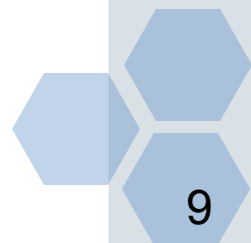




## 单总线运算器的结构形式 2— 数据通路

✧  $(R_i) \theta (R_j) \rightarrow R_k$  : 需要 3 步

序号	操作	发送的信号
1	$(R_i) \rightarrow LA$	$R_i \rightarrow IB, IB \rightarrow LA$
2	$(R_j) \rightarrow IB, ALU \text{ 运算, 结果} \rightarrow LC$	$R_i \rightarrow IB, C\_OP(\theta), ALU \rightarrow LC$
3	$(LC) \rightarrow R_k$	$LC \rightarrow IB, IB \rightarrow R_i$





### ❖如何完成运算操作:

- **双总线运算器的结构**  
形式 1：ALU+1 个寄存器

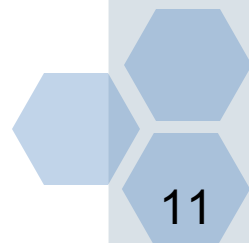




## 双总线运算器的结构形式 1— 数据通路

✧  $(R_i) \theta (R_j) \rightarrow R_k$  : 需要 2 步

序号	操作	发送的信号
1	$(R_i) \rightarrow IB1$ , $(R_j) \rightarrow IB2$ , A LU 运算, 结果 $\rightarrow LC$ ;	$R_i \rightarrow IB1$ , $R_i \rightarrow IB2$ , $C\_OP(\theta)$ , $ALU \rightarrow LC$
2	$(LC) \rightarrow R_k$	$LC \rightarrow IB1$ , $IB1 \rightarrow R_i$

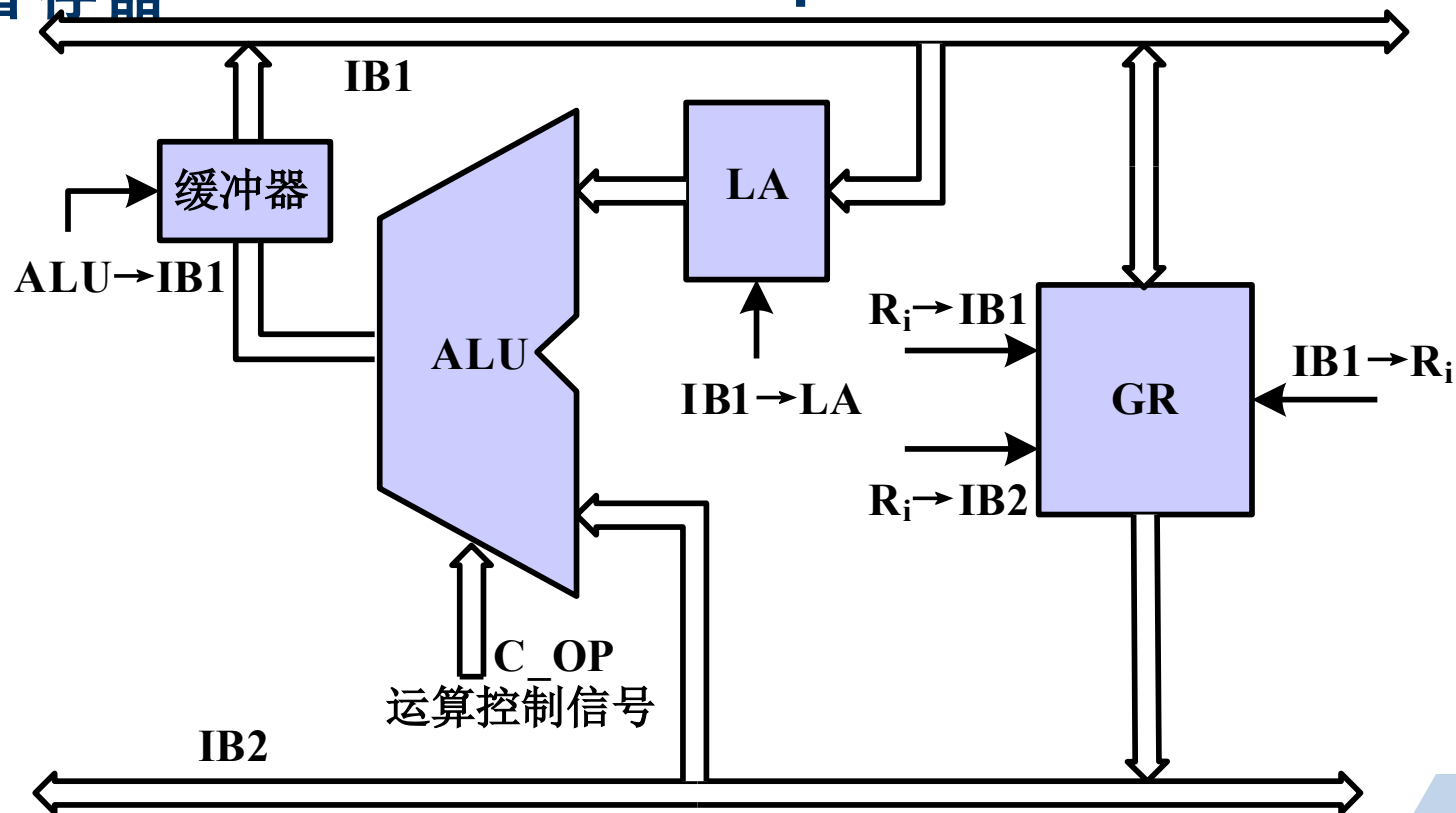




## 二、定点运算器的总线结构

- 双总线运算器的结构  
形式 2 : ALU+1 个  
暂存器

❖ 如何完成运算操作：  
 $(R_i) \theta (R_j) \rightarrow R_k$   
?





## 双总线运算器的结构形式 2— 数据通路

✧  $(R_i) \theta (R_j) \rightarrow R_k$  : 需要 2 步

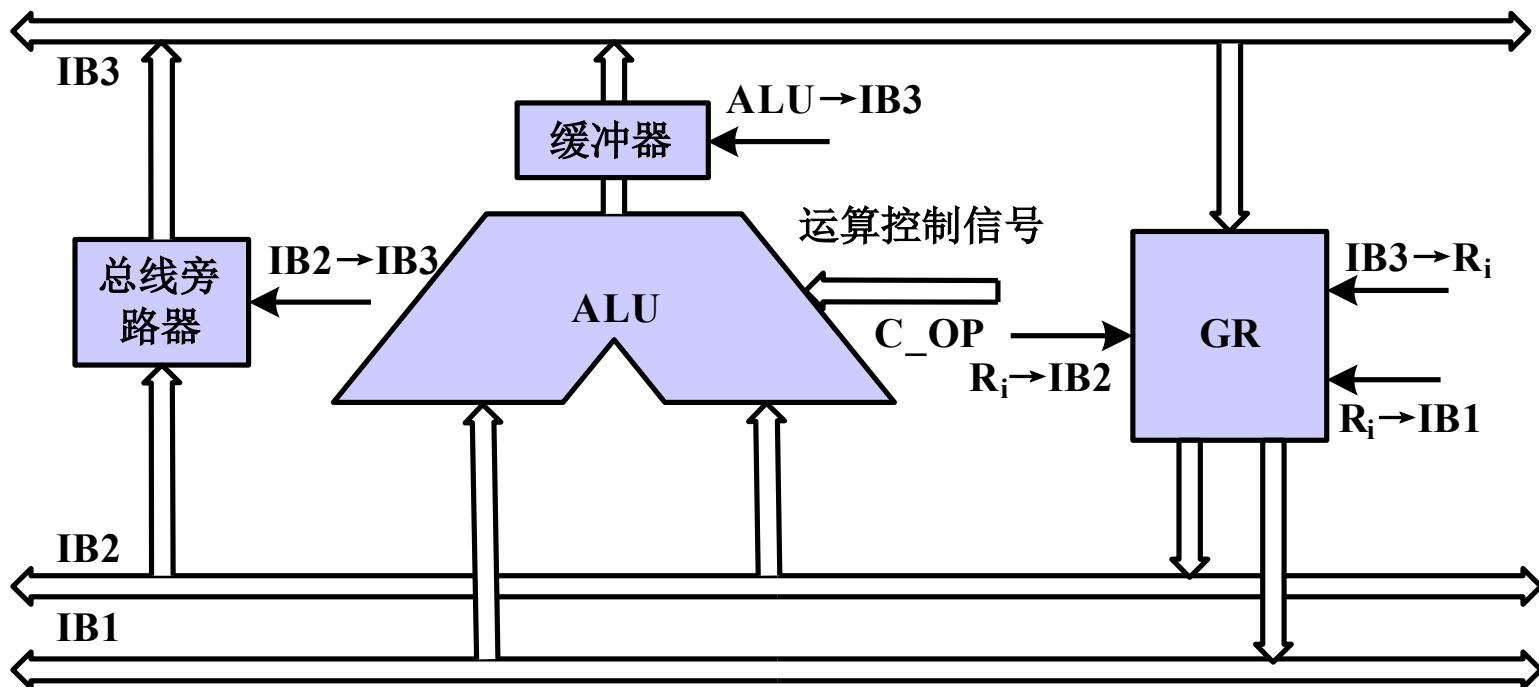
序号	操作	发送的信号
1	$(R_i) \rightarrow LA$	$R_i \rightarrow IB1, IB1 \rightarrow LA$
2	$(R_j) \rightarrow IB2,$ ALU 运 算, $IB1 \rightarrow R_k$ ;	$R_i \rightarrow IB2, C\_OP(\theta)$ , $ALU \rightarrow IB1,$ $IB1 \rightarrow R_i$



## 二、定点运算器的总线结构

### ❖ 3、三总线结构

❖ 如何完成运算操作：  
 $(R_i) \theta (R_j) \rightarrow R_k$   
?





## 三总线运算器的结构形式— 数据通路

❖  $(R_i) \theta (R_j) \rightarrow R_k$  : 需要 1 步

序号	操作	发送的信号
1	$(R_i) \rightarrow IB1,$ $(R_j) \rightarrow IB2,$ ALU 运算, $IB3 \rightarrow R_k$ ;	$R_i \rightarrow IB1,$ $R_j \rightarrow IB2,$ $C\_OP(\theta), ALU \rightarrow IB3,$ $IB3 \rightarrow R_i$

总结：分析数据通路的基本的原则：

在一个 CPU 周期（一步）内，某条总线上的数据必须是唯一的，且不能保留（至下一个 CPU 周期）。





## 三、标志寄存器

❖ 标志寄存器：又称为状态寄存器。

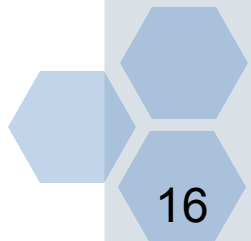
- 用来保存 ALU 操作结果的某些状态。
- 不同 CPU，标志寄存器中包含的标志也不尽相同。

❖ 最基本的 5 种运算结果标志：

① ZF：结果为零标志，

- 运算结果为全 0，ZF 置 1
- 运算结果不全为 0，ZF 置 0。

$$ZF = \overline{F_n + F_{n-1} + \dots + F_0}$$







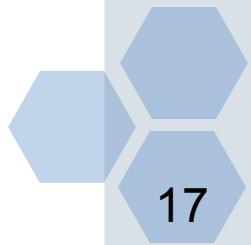
### 三、标志寄存器

#### ❖ 最基本的 5 种运算结果标志：

② **CF**：进位 / 借位标志位，**CF 标志只对无符号数运算有意义**

- 加法运算时：C=1 则 CF 置 1（表示有进位），否则置 0；
- 减法运算时：C=0 则 CF 置 1（表示不够减，有借位），否则置 0。

$$CF = ADD \bullet C + SUB \bullet \overline{C}$$





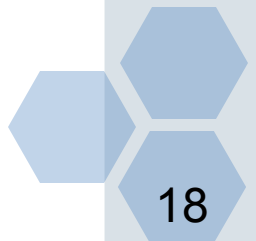
## 三、标志寄存器

### ❖ 最基本的 5 种运算结果标志：

③ **OF**：溢出标志，反映有符号数加减运算所得结果是否溢出； **OF** 标志只对带符号数运算有意义。

- 运算溢出：  $OF=1$
- 运算没有溢出：  $OF=0$

$$OF = C \oplus C'$$





## 三、标志寄存器

### ❖ 最基本的 5 种运算结果标志：

④**SF**：符号标志，记录运算结果的符号。

- 为运算结果的最高位
- 运算结果为正数时， $SF=0$ ，为负数  $SF=1$ 。

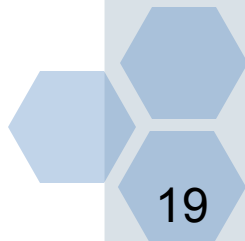
(无溢出)

$$SF = F_n$$

⑤**PF** 奇偶标志：反映运算结果中 “1” 的个数的奇偶性

- 当结果操作数中 “1” 的个数为偶数： $PF=1$
- 当结果操作数中 “1” 的个数为奇数： $PF=0$

$$PF = F_n \oplus F_{n-1} \oplus \dots \oplus F_0$$





### 三、标志寄存器（举例）

MOV AL, 80H ; 80H→AL

ADD AL, 80H ; (AL)

+ (80H) → AL

$$\begin{array}{r} 1000\ 0000 \\ +\ 1000\ 0000 \\ \hline 0000\ 0000 \end{array}$$

❖ ZF=1

❖ CF=1

❖ OF=1

❖ 执行结果：(AL) = 00H ;

❖ SF=0

❖ PF=1





The End !