第4章 运算方法与运算器

BACK

4.1 定点数的加减运算及实现

- 一 补码加减运算与运算器
- 二 机器数的移位运算
- 三 移码加减运算与判溢
- 四 十进制加法运算

一、补码加减运算与运算器

1、补码加减运算方法

❖ 补码的**加法**运算公式：

- $[X+Y]_{补} = [X]_{补} + [Y]_{补}$

❖ 补码的**减法**运算公式：

- $[X-Y]_{补} = [X+（-Y）]_{补}$
  $= [X]_{补} + [-Y]_{补}$

证明：

$$[X]_{补} = 2^{n+1} + X \quad (\mathrm{mod}\ 2^{n+1})$$

$$[Y]_{补} = 2^{n+1} + Y \quad (\mathrm{mod}\ 2^{n+1})$$

$$\begin{aligned}[X]_{补} + [Y]_{补} &= 2^{n+1} + X + 2^{n+1} + Y \quad (\mathrm{mod}\ 2^{n+1}) \\ &= 2^{n+1} + (X + Y) \quad (\mathrm{mod}\ 2^{n+1}) \\ &= [X+Y]_{补} \quad (\mathrm{mod}\ 2^{n+1})\end{aligned}$$

1、补码加减运算方法

- 补码的加减运算的公式是：
  - $[X+Y]_{补} = [X]_{补} + [Y]_{补}$
  - $[X-Y]_{补} = [X]_{补} + [-Y]_{补}$
- 特点：
  - 使用补码进行加减运算，**符号位和数值位一样参加运算**。
  - **补码的减法可以用加法来实现**，任意两数之差的补码等于被减数的补码与减数相反数的补码之和 。
- **注意**：该公式不适合任何其他机器数编码（原码、反码、移码）。

求补运算：$[Y]_{补} \rightarrow [-Y]_{补}$

- 求补规则：**将$[Y]_{补}$包括符号位在内每一位取反，末位加1。**
- 若$[Y]_{补} = Y_0，Y_1\ldots\ldots Y_n$，则：

$$[-Y]_{补} = \overline{Y_0}\,\overline{Y_1}\cdots\cdots\overline{Y_n} + 1$$

- 若$[Y]$补 $= Y_0 . Y_1\ldots\ldots Y_n$，则：

$$[-Y]_{补} = \overline{Y_0}\,\overline{Y_1}\cdots\cdots\overline{Y_n} + \mathbf{0.0}\cdots\cdots\mathbf{0}1$$

- 例：　$[X]_{补}$ =0.1101，则：　$[-X]_{补}$ =　1.0011
- 　　　$[Y]_{补}$ =1.1101，则：　$[-Y]_{补}$ =　0.0011

补码加减运算举例

❖例：已知X=+1011，Y=-0100，用补码计算X+Y和X-Y。

- 写出补码：

  $[X]_{补}$ =0, 1011　　　$[Y]_{补}$ =1, 1100

  $[-Y]_{补}$ =0, 0100

- 计算：

$$\begin{array}{r} 0,1011 \\ +\ 1,1100 \\ \hline 0,0111 \end{array} \qquad \begin{array}{r} 0,1011 \\ +\ 0,0100 \\ \hline 0,1111 \end{array}$$

$[X+Y]_{补}$ = 0, 0111　　　$[X-Y]_{补}$ = 0, 1111

练习

1、A=0.1011，B=-0.1110，求A+B

2、A=0.1011，B=-0.0010，求A-B

补码加法示例

例3：A=0.1011，B=-0.1110，求A+B

$\because [A]_{补}=0.1011 \quad [B]_{补}=1.0010$

$$
\begin{array}{rl}
 0.1011 & [A]_{补} \\
+\ 1.0010 & [B]_{补} \\
\hline
 1.1101 & [A+B]_{补}
\end{array}
$$

$\therefore [A+B]_{补}=1.1101$

A+B=-0.0011

补码减法示例

例4：A=0.1011，B=-0.0010，求A-B

$\because [A]_{补}=0.1011 \quad [B]_{补}=1.1110$

$[-B]_{补}=0.0010$

$$
\begin{array}{rl}
 & 0.1011 \quad [A]_{补} \\
+ & 0.0010 \quad [-B]_{补} \\
\hline
 & 0.1101 \quad [A-B]_{补}
\end{array}
$$

$\therefore [A-B]_{补}=0.1101$

$A-B=0.1101$

2、补码加减运算的溢出判断

- 例：X=+1000，Y=+1001，求X+Y；
- 当运算结果超出机器数的表示范围时，称为溢出。计算机必须具备检测运算结果是否发生溢出的能力，否则会得到错误的结果。
- 对于加减运算，可能发生溢出的情况：同号（两数）相加，或者异号（两数）相减。
- 确定发生溢出的情况：
  - 正数相加，且结果符号位为1；
  - 负数相加，且结果符号位为0；
  - 正数－负数，且结果符号位为1；
  - 负数－正数，且结果符号位为0；

常用的判溢方法（补码加减运算）

（1）单符号位判溢方法

$\overline{ADD}/SUB$ =0：做加法；$\overline{ADD}/SUB$ =1：做减法

$[X]_{补} = X_f\ X_1 \cdots\cdots X_n$

$[Y]_{补} = Y_f\ Y_1 \cdots\cdots Y_n$

$[X \pm Y]_{补} = S_f\ S_1 \cdots\cdots S_n$

$$V = \overline{\overline{ADD}/SUB}\,(\overline{X_f}\,\overline{Y_f}\,S_f + X_f\,Y_f\,\overline{S_f}) + \overline{ADD}/SUB\,(\overline{X_f}\,Y_f\,S_f + X_f\,\overline{Y_f}\,\overline{S_f})$$

V=1:溢出

正数相加，且结果符号位为1

负数相加，且结果符号位为0

正数－负数，且结果符号位为1

负数－正数，且结果符号位为0

常用的判溢方法（补码加减运算）

补码加减运算符号位及进位的真值表

	$\overline{ADD}/SUB$	X_f	Y_f	C_1	S_f	C_f	V	说明
加法	0	0	0	0	0	0	0	无溢出
	0	0	0	1	1	0	1	正溢出
	0	1	1	0	0	1	1	负溢出
	0	1	1	1	1	1	0	无溢出
减法	1	0	1	0	0	0	0	无溢出
	1	0	1	1	1	0	1	正溢出
	1	1	0	0	0	1	1	负溢出
	1	1	0	1	1	1	0	无溢出

最高有效位运算产生的进位

符号位运算产生的进位

常用的判溢方法（补码加减运算）

（2）进位判溢方法

- 当**最高有效位产生的进位和符号位产生的进位不同**时，加减运算发生了溢出。
- $V = C_1 \oplus C_f$

❖ 例： X=+1000，Y=+1001，求X+Y；

❖ $[X]_{补}$ =0,1000　　　$[Y]_{补}$ =0,1001

$$
\begin{array}{rl}
 & 0\ \ 1000 \\
+ & 0\,.\,1001 \\
\hline
 & 1\ \ 0001
\end{array}
$$

C_1=1，C_f=0：溢出

常用的判溢方法（补码加减运算）

（2）双符号位判溢方法

- X和Y采用双符号位补码参加运算，正数的双符号位为00，负数的双符号位为11；当**运算结果的两位符号S_{f1} S_{f2}不同时**（01或10），发生溢出。

$$
\begin{array}{rl}
 & X_f\,X_f\;\;X_1\;X_2\;\ldots\ldots X_n \\
+ & Y_f\,Y_f\;\;Y_1\;Y_2\;\ldots\ldots Y_n \\
\hline
 & S_{f1}\,S_{f2}\;\;S_1\;\;S_2\;\;\ldots\ldots S_n
\end{array}
$$

- $V = S_{f1} \oplus S_{f2} = X_f \oplus Y_f \oplus C_f \oplus S_f$
- S_{f1} S_{f2}=01，则正溢出；S_{f1} S_{f2}=10，则负溢出。

双符号位判溢方法举例

❖ 例：用补码计算X+Y和X-Y

- （1）X=+1000，Y=+1001
- （2）X=-1000，Y=1001

	$[X]_补$	00，1000
+	$[Y]_补$	00，1001
	$[X+Y]_补$	01，0001

$S_{f1}\ S_{f2}$=01，正溢出

	$[X]_补$	00，1000
+	$[-Y]_补$	11，0111
	$[X-Y]_补$	11，1111

$S_{f1}\ S_{f2}$=11，无溢出

	$[X]_补$	11，1000
+	$[Y]_补$	00，1001
	$[X+Y]_补$	00，0001

$S_{f1}\ S_{f2}$=00，无溢出

	$[X]_补$	11，1000
+	$[-Y]_补$	11，0111
	$[X-Y]_补$	10，1111

$S_{f1}\ S_{f2}$=10，负溢出

练习

已知 A＝－1011B，B＝－0111B，求$[A+B]_{补}$。

已知 A＝＋0100 0010B，B＝＋0011 0011B，试用双符号法计算 A+B。

练习

[例 18] 已知 A＝－1011B，B＝－0111B，求$[A+B]_{补}$。

[解] A＝－1011B，$[A]_{补}=1.0101B$

B＝－0111B，$[B]_{补}=1.1001B$

$[A+B]_{补}=[A]_{补}+[B]_{补}=1.0101B+1.1001B=0.1110B$

$$
\begin{array}{r}
1\ 0101 \\
+\quad 1\ 1001 \\
\hline
1\ 0\ 1110
\end{array}
$$

丢弃

两个操作数的符号均为 1，结果为 0，产生溢出。

练习

[例 20] 已知 A=+0100 0010B，B=+0011 0011B，试用双符号法计算 A+B。

[解] A=+0100 0010B，$[A]_{补}=\boxed{0\ 0}100\ 0010B$

B=+0011 0011B，$[B]_{补}=\boxed{0\ 0}011\ 0011B$

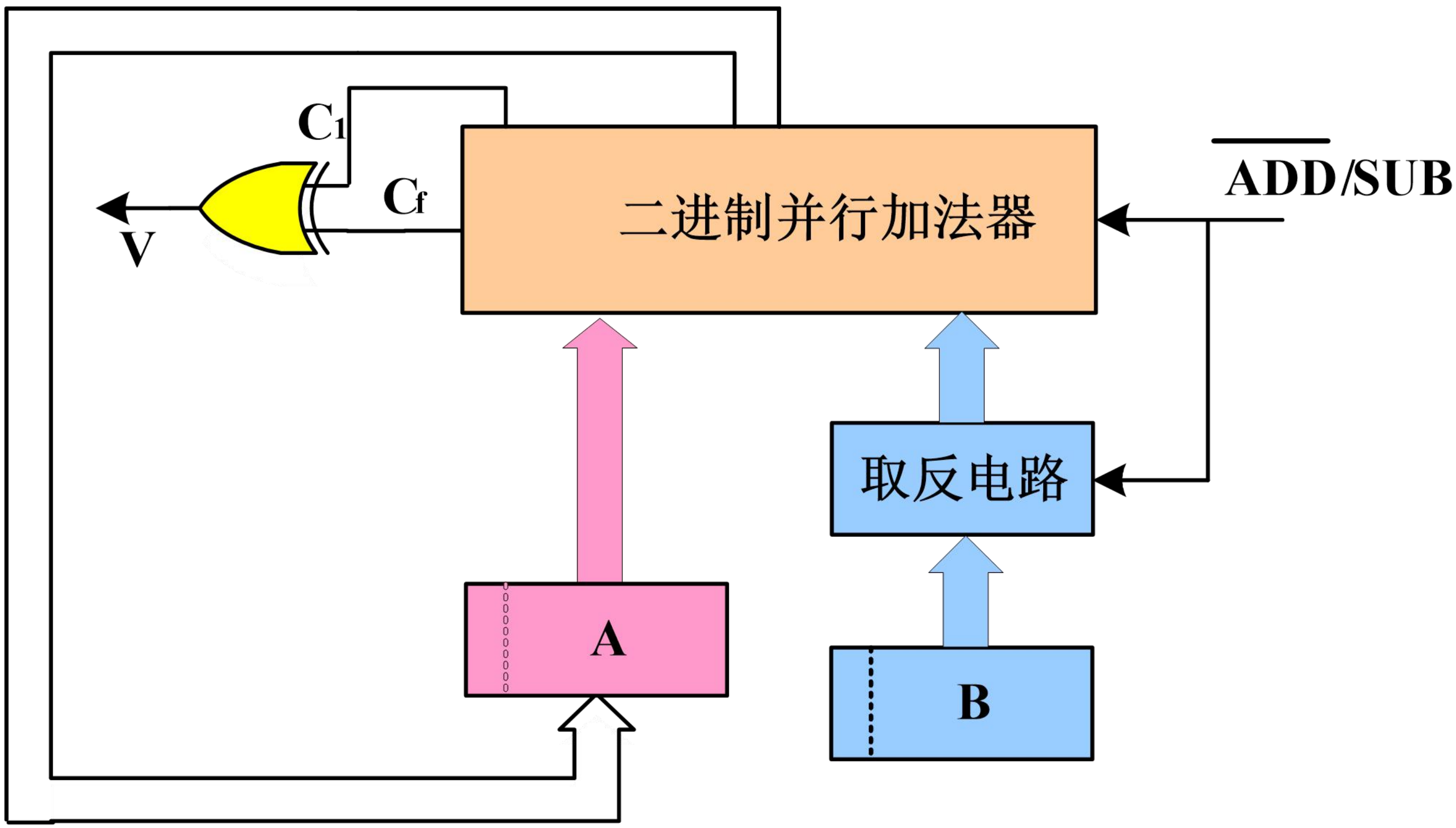
$[A+B]_{补}=[A]_{补}+[B]_{补}=0\ 0100\ 0010B+0\ 011\ 0011B$

$=0\ 0111\ 0101B$

$$
\begin{array}{r}
0\ \ 0100\ \ 0010 \\
+\ 0\ \ 0011\ \ 0011 \\
\hline
0\ \ 0111\ \ 0101 \\
C_f\,C\qquad\qquad\quad
\end{array}
$$

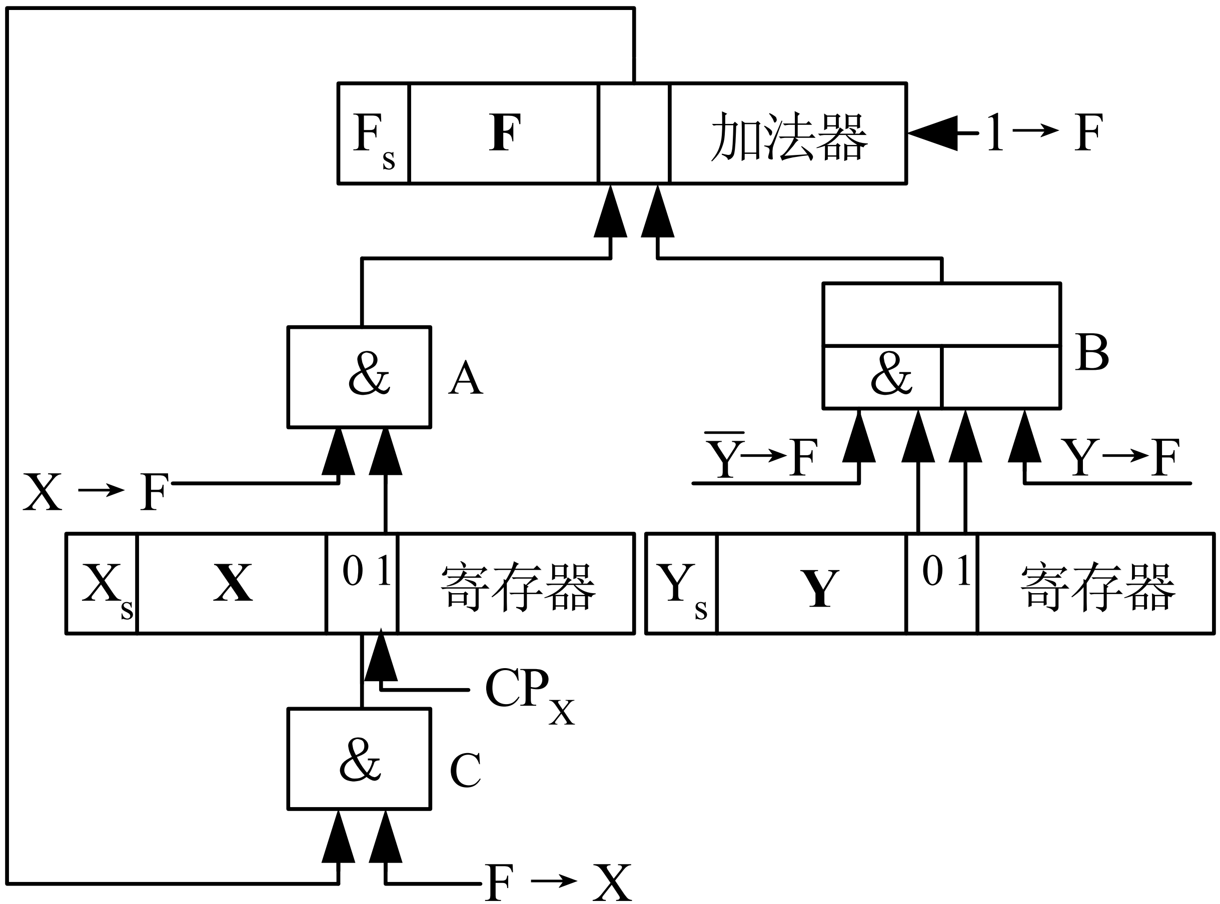
两个符号位相同，没有产生溢出。

3、补码加减运算器



3、补码加减运算器的实现

- **核心部件**：一个普通的二进制并行加法器。
- **A**：累加器，存放$[X]_{补}$；**B**：寄存器，存放$[Y]_{补}$；
- **取反电路**:
- $\overline{ADD}/SUB$ =0时，补码加法器，将B寄存器直接送入并行加法器；
- $\overline{ADD}/SUB$ =1时，补码减法器，将B取反送入并行加法器，同时，并行加法器的最低位产生进位，即B取反加1，此时并行加法器的运算相当于$[A]_{补}$加$[-B]_{补}$，完成减法运算。



补码加减运算器框图

The End!