

# **Тема: Арифметические основы организации ЭВМ**



- 1. Сложение и вычитание чисел с фиксированной запятой**
- 2. Умножение и деление чисел с фиксированной запятой**
- 3. Десятичные операции**



# Сложение и вычитание чисел с фиксированной запятой

Сложение и вычитание представляют пару операций «типа сложения», т.е. алгебраическое сложение, которое, в свою очередь, можно понимать как сложение чисел со знаком, заданных в обратном или дополнительном коде.

Вычитание может выполняться непосредственно (с использованием, например, специальных операционных элементов – вычитателей) или косвенно, путем сведения его к сложению:

$$Z := X - Y = X + (-Y)$$

# Сложение и вычитание чисел с фиксированной запятой

В последнем случае достаточно, как видно, изменить знак второго операнда. Если операнды (и результат) представлены в дополнительном коде, изменение знака производится путем инверсии **всех** разрядов и добавления 1 в младшем разряде.

**Например,**

$$\begin{array}{r} Y = 5 \sim 0.101_2 \\ -Y = -5 \sim 1\ 010_2 \\ + \quad \quad \quad 1 \\ \hline 1.011_2 = (-5)_{\text{доп}}, \end{array}$$

и наоборот,

$$\begin{array}{r} Y = -5 \sim 1.011_2 \\ -Y = 5 \sim 0\ 100_2 \\ + \quad \quad \quad 1 \\ \hline 0.101_2 = (5)_{\text{доп}}. \end{array}$$

# Сложение и вычитание чисел с фиксированной запятой

## Правила алгебраического сложения чисел в обратном и дополнительном кодах

тривиальны: обратные или дополнительные коды операндов суммируются как обыкновенные числа без знака, возможная единица переноса из знакового разряда (старшего знакового разряда, если код модифицированный) циклически переносится в младший разряд для второго суммирования (обратный код) или отбрасывается (дополнительный код).

« + » – знак операции сложения с циклическим переносом (Пример 1).

# Сложение и вычитание чисел с фиксированной запятой

## Пример 1.

$$\begin{array}{r} X = -5 \\ + \\ Y = 7 \\ \hline Z = 2 \end{array}$$

$$\begin{array}{r} X_{\text{обр}} = 1.010_2 \\ + \\ Y_{\text{обр}} = 0.111_2 \\ \hline 10.001 \\ + \quad 1 \\ \hline Z_{\text{обр}} = 0.010_2 \end{array}$$

# Сложение и вычитание чисел с фиксированной запятой

## Пример 2.

$$\begin{array}{r} X = 5 \\ + \\ Y = -7 \\ \hline Z = -2 \end{array}$$

$$\begin{array}{r} X_{\text{доп мод}} = 00.101_2 \\ + \\ Y_{\text{доп мод}} = 11.001_2 \\ \hline Z_{\text{доп мод}} = 11.110_2 \end{array}$$



# Сложение и вычитание чисел с фиксированной запятой

В обратном коде отсутствие выходного переноса свидетельствует о неположительном результате (Пример 4), а наличие его – о результате положительном (Пример 1).

Пример 4.

$$\begin{array}{r} X = 5 \\ + \\ Y = -5 \\ \hline Z = 0 \end{array}$$

$$\begin{array}{r} X_{\text{обр}} = 0.101_2 \\ + \\ Y_{\text{доп}} = 1.010_2 \\ \hline Z_{\text{обр}} = 1.111_2 = (-0)_{\text{обр}} \\ p_{\text{вых}} = 0 \end{array}$$



# Сложение и вычитание чисел с фиксированной запятой

Обнаружение переполнения разрядной сетки при сложении может производиться несколькими способами.

**Самый простой способ – использование модифицированного кода** (с двумя знаковыми разрядами).

Старший знаковый разряд даже при переполнении сохраняет информацию о знаке результата («Разряд знака»).

Младший – «Разряд переполнения».  
Комбинация знаков при «положительном» переполнении – 01,  
при «отрицательном» – 10.

# Сложение и вычитание чисел с фиксированной запятой

## Пример 5.

$$\begin{array}{r} X = 3 \\ + \\ Y = 6 \\ \hline Z = 9 > 7 \end{array}$$

$$\begin{array}{r} X_{\text{доп мод}} = 00.011_2 \\ + \\ Y_{\text{доп мод}} = 00.110_2 \\ \hline Z_{\text{доп мод}} \sim 01.001_2 \text{ (положительное} \\ \text{переполнение)} \end{array}$$

## Пример 6.

$$\begin{array}{r} x = -3 \\ + \\ y = -6 \\ \hline z = -9 < 8 \end{array}$$

$$\begin{array}{r} X_{\text{доп мод}} = 11.101_2 \\ + \\ Y_{\text{доп мод}} = 11.010_2 \\ \hline Z_{\text{доп мод}} \sim 10.111_2 \text{ (отрицательное} \\ \text{переполнение)} \end{array}$$

# Сложение и вычитание чисел с фиксированной запятой

В примере 6 указано граничное значение (-8), которое может быть представлено без переполнения:

$$(-8)_{\text{доп мод}} = 11.000_2$$

Недостаток способа модифицированного кода — расширение разрядной сетки на один разряд.

# Сложение и вычитание чисел с фиксированной запятой

**Второй способ** обнаружения переполнения - сравнение переносов в знаковый разряд и из знакового разряда. **Переполнение** - при несовпадении этих переносов. Фактически здесь тоже «задействован» модифицированный дополнительный код.

Случай А. Неотрицательные операнды.

	0	0.
	+0	0.
	0	0.
<hr/>		
Перенос «в»	0	X
Сумма первичная	0	0.
Сумма конечная	0	X.

# Сложение и вычитание чисел с фиксированной запятой

Правило сравнения переносов дает значение признака переполнения:

$$\varphi_p = 0 \oplus X = X$$

(переполнение при  $X = 1$ ).

Слева от штриховой черты показаны значения воображаемого модифицированного дополнительного кода.

Правило этого способа дает такое же значение признака переполнения:

$$\varphi_m = 0 \oplus X = X$$

# Сложение и вычитание чисел с фиксированной запятой

## Случай В. Отрицательные операнды.

	+1	1.
	1	1.
Перенос «в»	1	X
Сумма первичная	0	0.
Сумма конечная	1	X.

Здесь тоже  $\varphi_p = \varphi_m = 1 \oplus X = X$  —

(переполнение может быть только отрицательное - при отсутствии переноса из старшего цифрового разряда).

# Сложение и вычитание чисел с фиксированной запятой

## Случай С. Операнды имеют разные знаки

	+0 1	0. 1.
Перенос «в»	X	X
Сумма первичная	1	0.
Сумма конечная	$\overline{X}$	$\overline{X.}$

Оба признака переполнения снова совпадают, они имеют нулевые значения (переполнение в принципе невозможно):

$$\varphi_p = X \oplus X = 0$$

$$\varphi_m = X \oplus X = 0$$

# Сложение и вычитание чисел с фиксированной запятой

Третий способ - сравнение знаков. Реализуется программно (микропрограммно). Сначала проверяется, имеют ли операнды одинаковые знаки. И, если имеют, совпадает ли с этими знаками знак результата. Переполнение соответствует несовпадению (рис. 4).

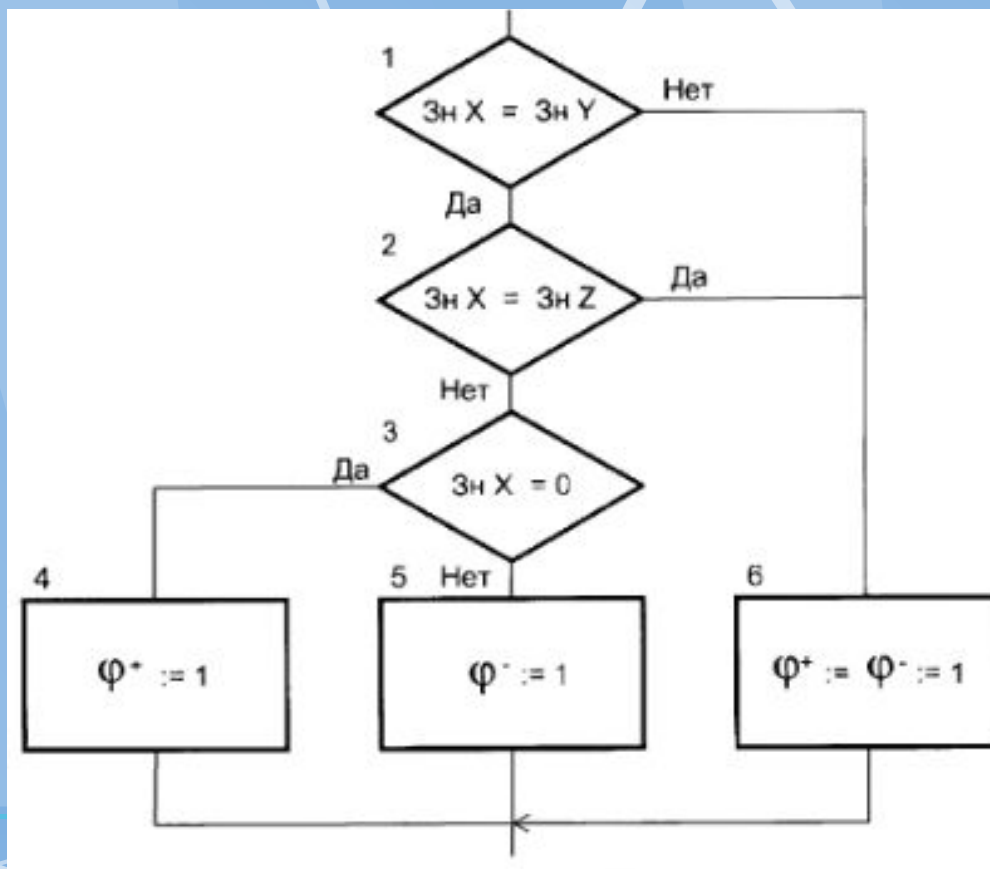


Рис. 4. Обнаружение переполнения