

XI. Операционные устройства ЭВМ

-Целочисленные арифметико-логические устройства: устройства выполнения логических операций, устройства целочисленного сложения/вычитания, устройства целочисленного умножения, устройства целочисленного деления.

Устройства обработки чисел с плавающей запятой: устройства сложения/вычитания, устройства умножения, устройства деления, устройства вычисления функций.

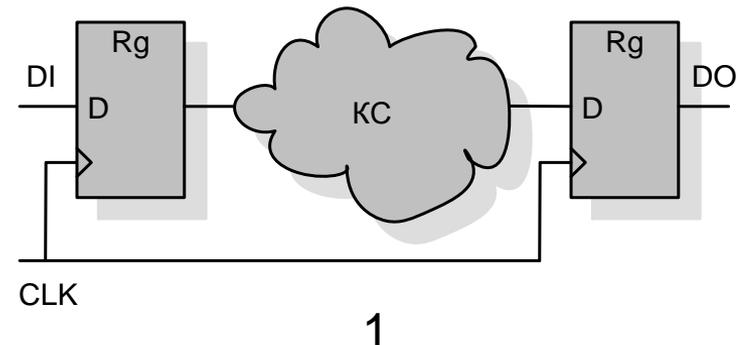
- Устройства SSE арифметики.

-Операционные устройства состоят из:

Регистров для хранения данных;

Шин передачи данных;

Комбинационных схем реализации функций;



Устройства целочисленного сложения/вычитания

- Накапливающие сумматоры (последовательные).
- Параллельные сумматоры: с последовательным переносом, с параллельным переносом, с условным переносом, с групповой структурой.

Схема последовательного сумматора

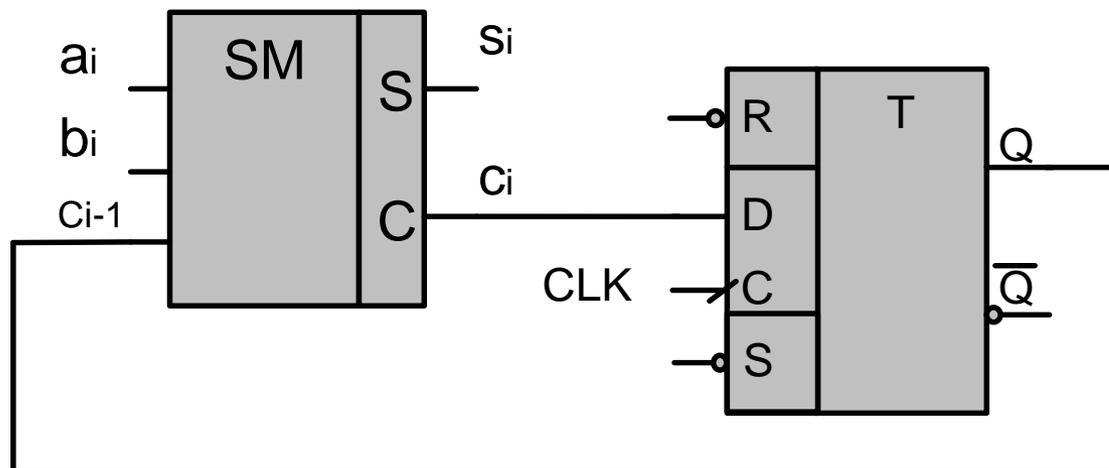


Схема параллельного сумматора с последовательным переносом

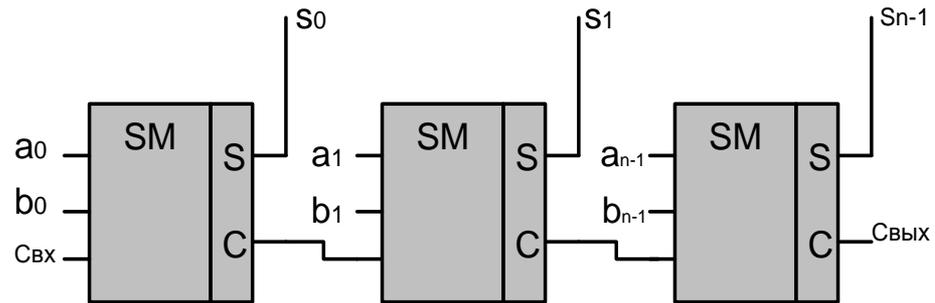
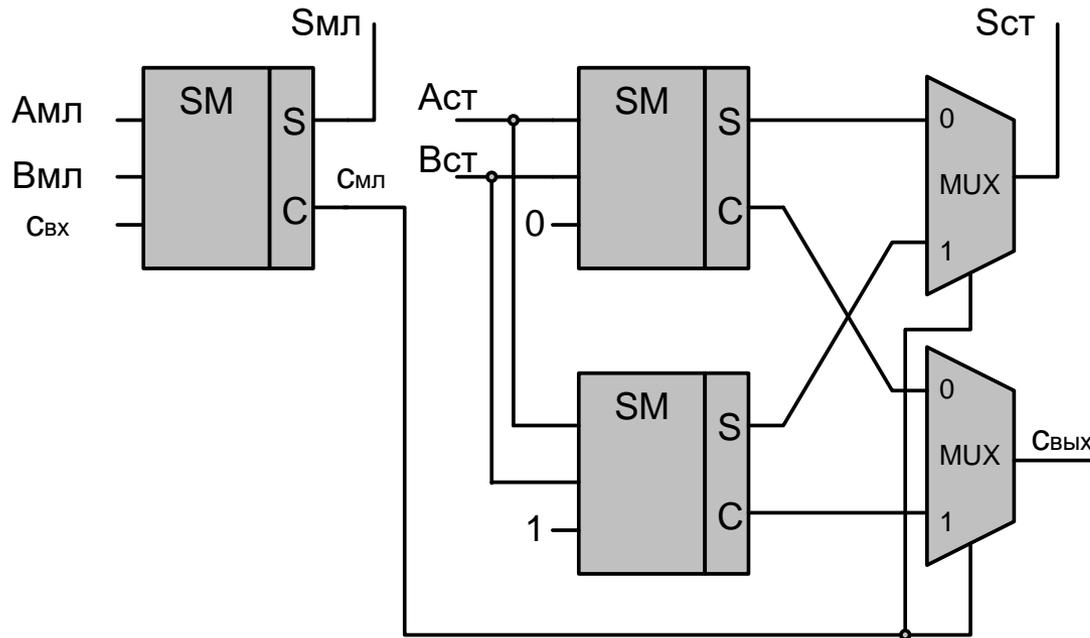
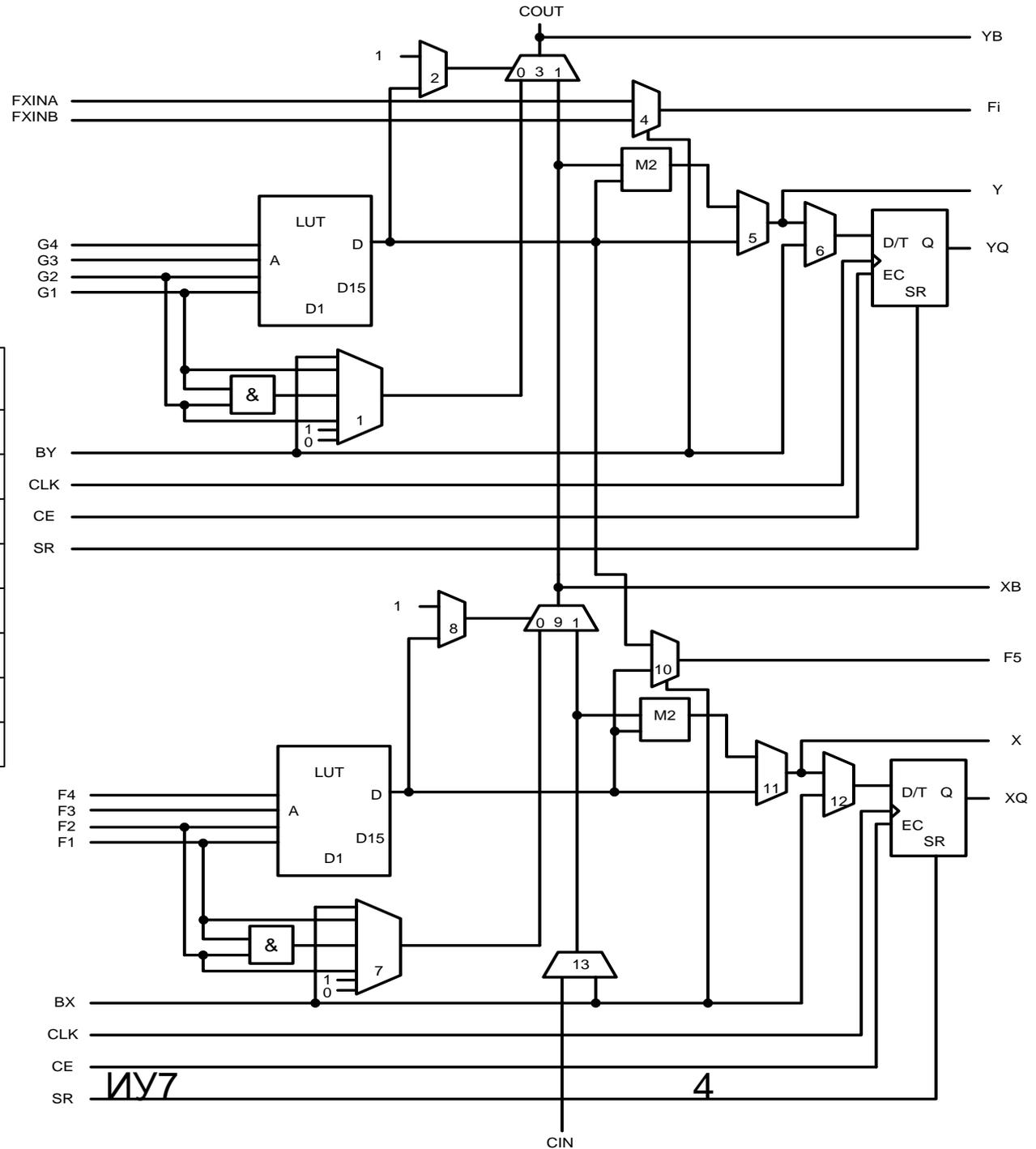


Схема сумматора с условным переносом



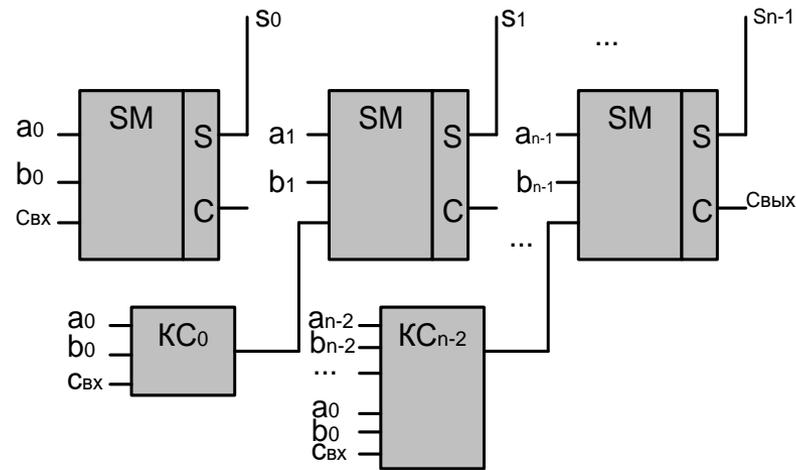
Структура блока типа SLICEL

	a	b	LUT	S	C
0	0	0	0	0	ab
0	0	1	1	1	C'
0	1	0	1	1	C'
0	1	1	0	0	ab
1	0	0	0	1	ab
1	0	1	1	0	C'
1	1	0	1	0	C'
1	1	1	0	1	ab

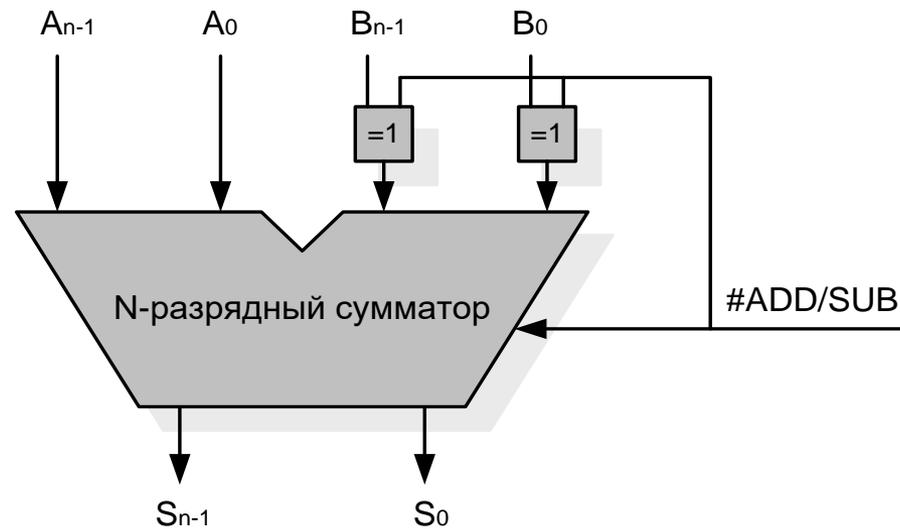


Архитектура ЭВМ

Схема параллельного сумматора с параллельным переносом



Устройство целочисленного сложения/вычитания



Устройства целочисленного умножения

Умножение сводится к последовательному формированию частных произведений и их сложению.

По способу формирования частных произведений:

умножение со старших разрядов множителя со сдвигом влево, умножение с младших разрядов множителя со сдвигом вправо.

По способу накопления частных произведений: матричные умножители, древовидные умножители.

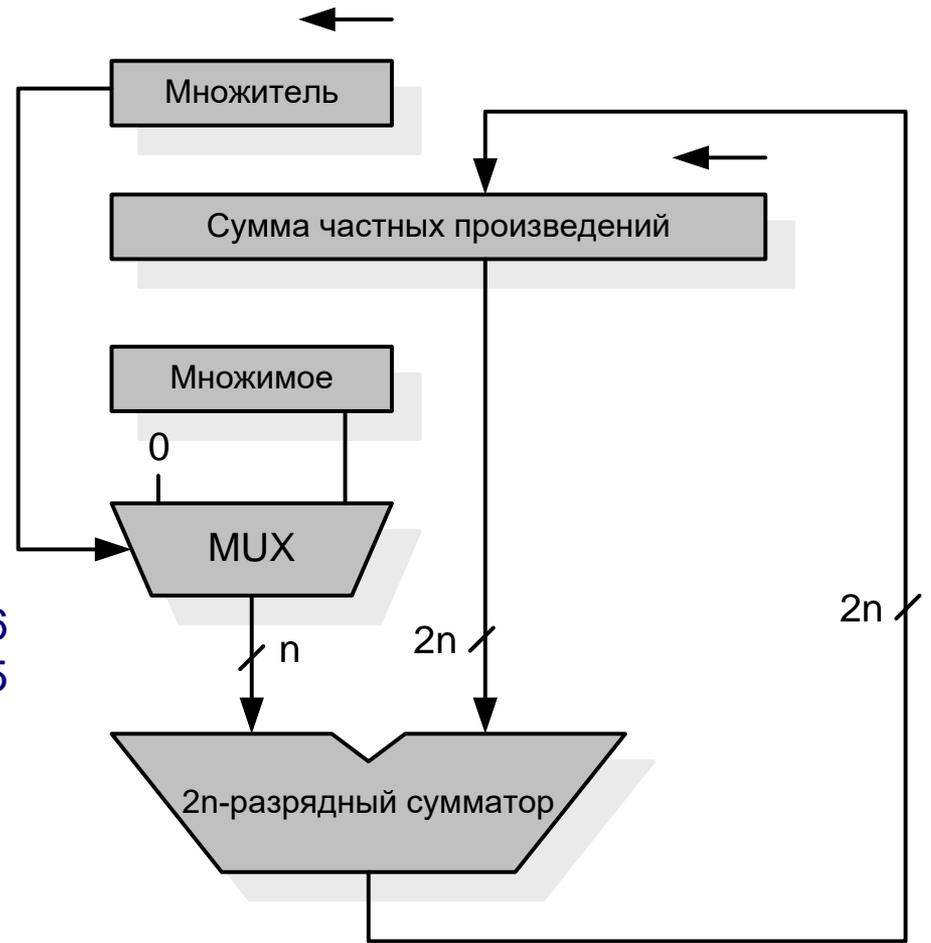
Способы ускорения работы устройств умножения:

- сокращение количества частных произведений;
- обработка нескольких разрядов множителя за такт;
- параллельное вычисление нескольких СЧП;
- конвейеризация умножителей.

Умножение со старших разрядов множителя со сдвигом влево

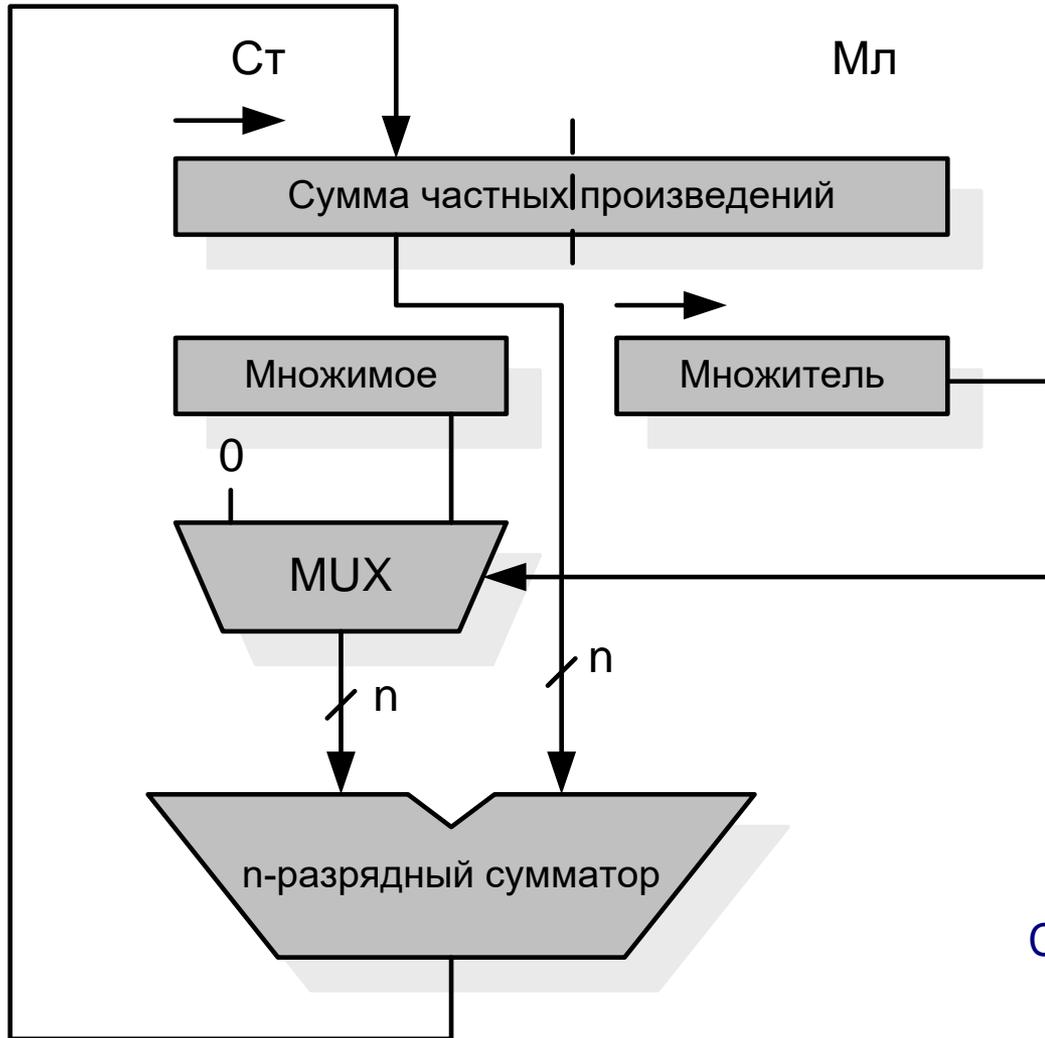
Старший разряд множителя определяет очередное частное произведение (ЧП), которое складывается с накопленной суммой частных произведений (СЧП). После этого СЧП и множитель сдвигаются на один разряд влево.

	A	1 1 0	6
x	B	1 0 1	5
<hr/>			
	ЧП0	1 1 0	
	СЧП0	1 1 0	
	<-СЧП0	1 1 0 0	
<hr/>			
	ЧП1	0 0 0	
	СЧП1=СЧП0+ЧП1	1 1 0 0	
	<-СЧП1	1 1 0 0 0	
<hr/>			
	ЧП2	1 1 0	
	СЧП2=СЧП1+ЧП2	1 1 1 1 0	30



(-) 2-n разрядный сумматор и шины данных.

Умножение с младших разрядов множителя со сдвигом вправо

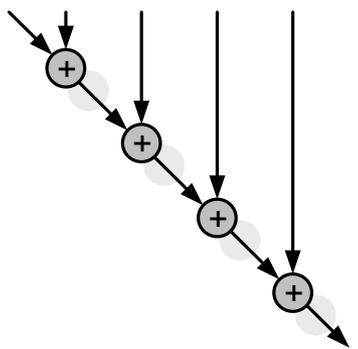
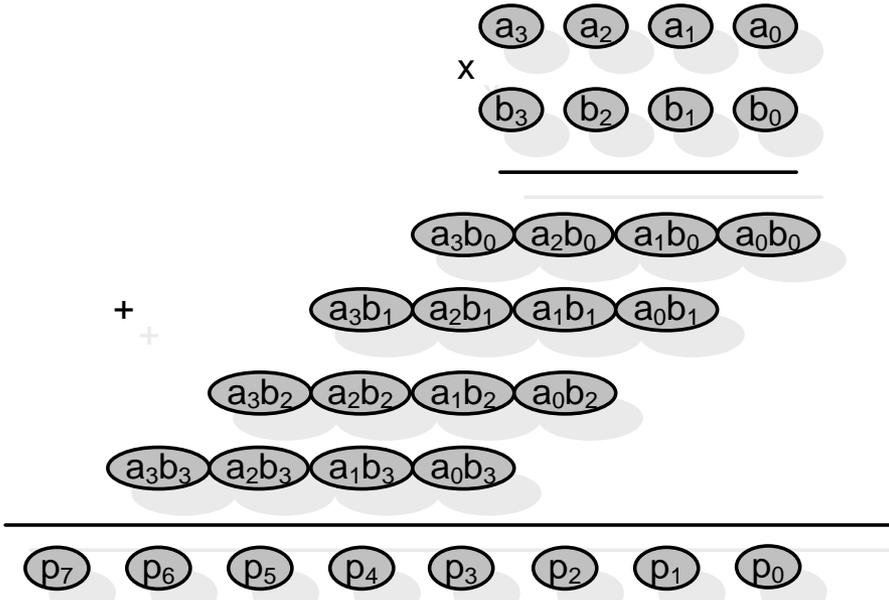


Младший разряд множителя определяет очередное частное произведение (ЧП), которое складывается с накопленной суммой частных произведений (СЧП). После этого СЧП и множитель сдвигаются на один разряд вправо.

	A	1	1	0	6	
x	B	1	0	1	5	
	ЧП0	1	1	0		
	СЧП0	1	1	0		
	СЧП0->	0	1	1	0	
	ЧП1	0	0	0		
	СЧП1=СЧП0+ЧП1	0	1	1	0	
	СЧП1->	0	0	1	1	0
	ЧП2	1	1	0		
	СЧП2=СЧП1+ЧП2	1	1	1	1	0
					30	

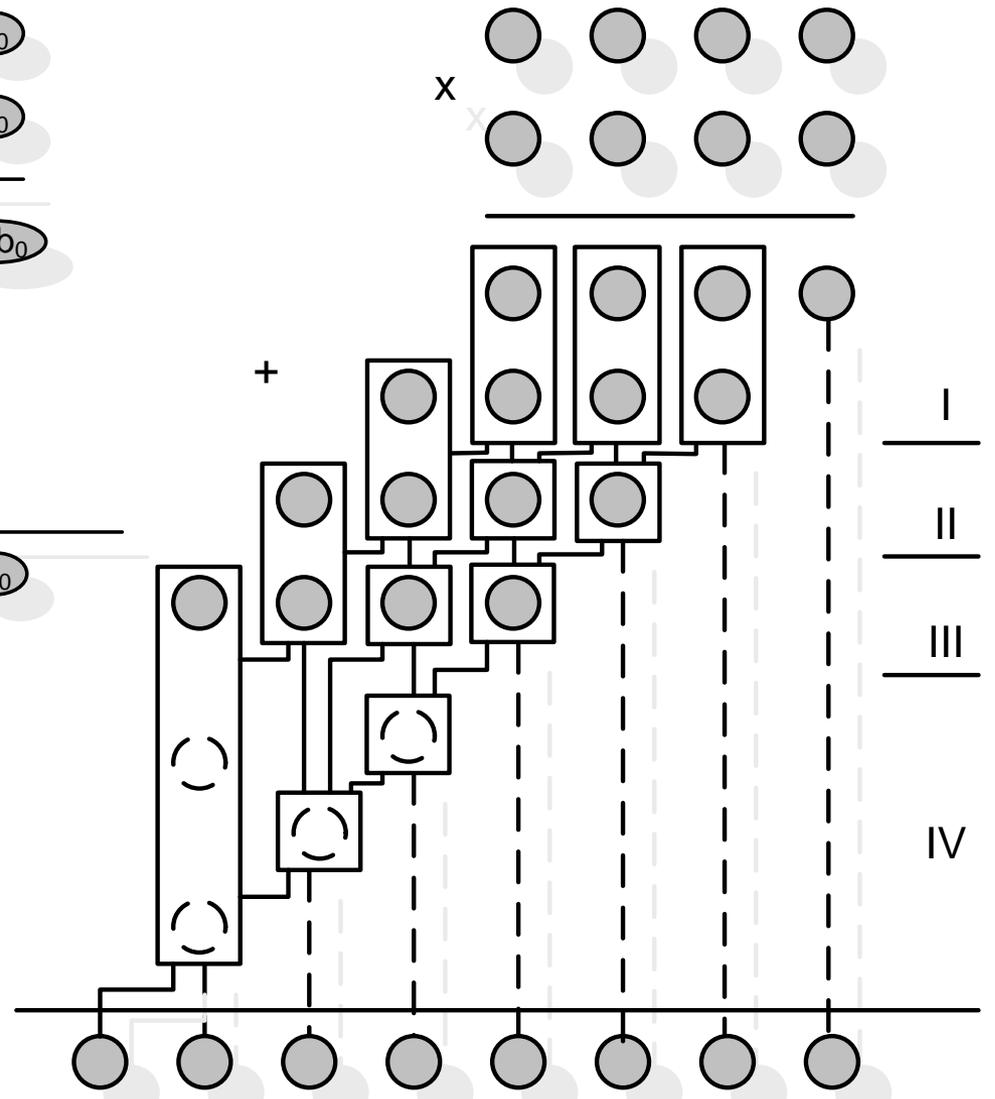
(+) n-разрядный сумматор и шины данных.

Матричные умножители



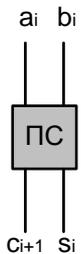
Архитектура ЭВМ

ИУ7

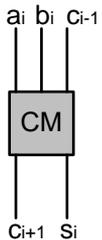


9

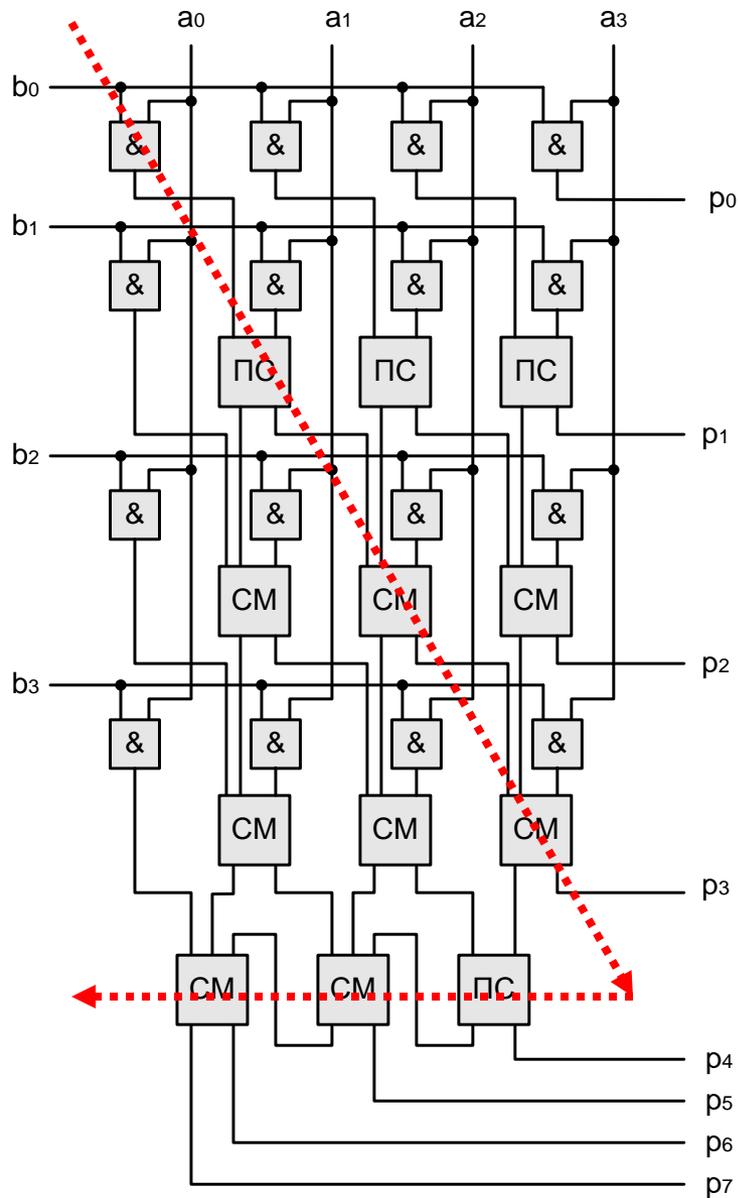
Матричные умножители



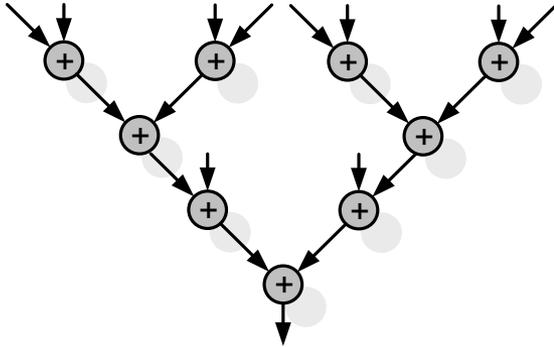
Полусумматор



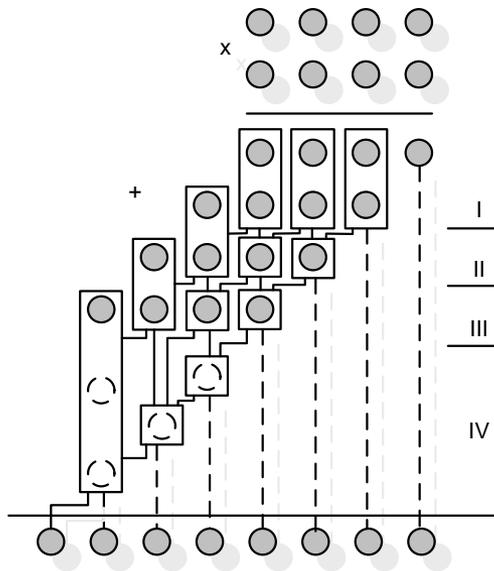
Сумматор



Древовидные умножители (схема Уоллеса)

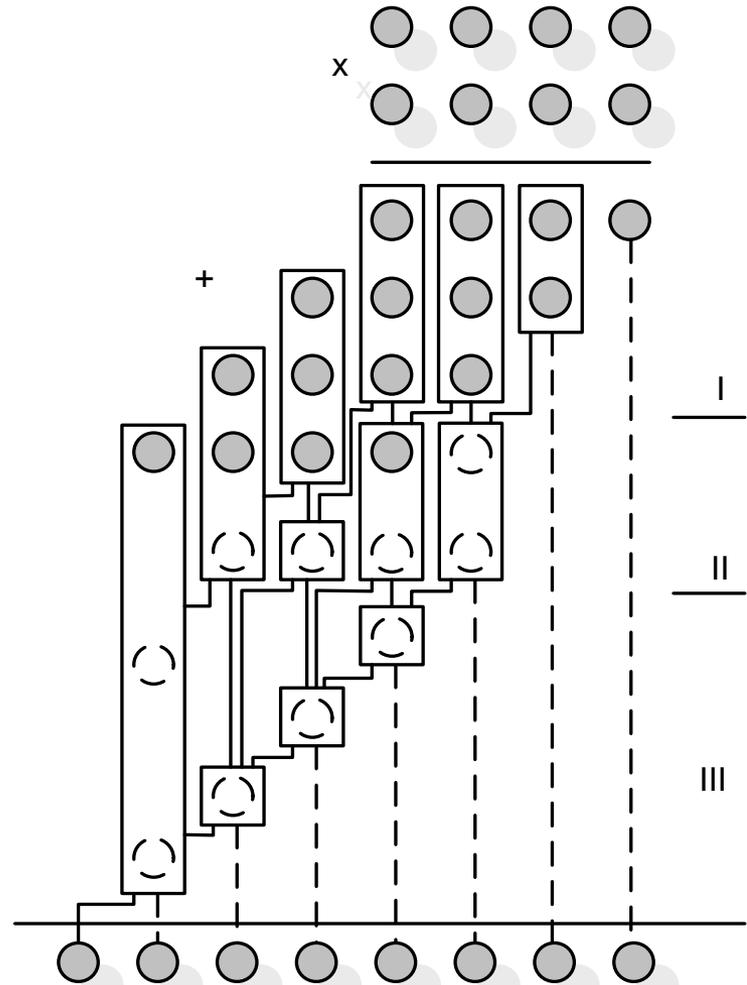


Матричные умножители



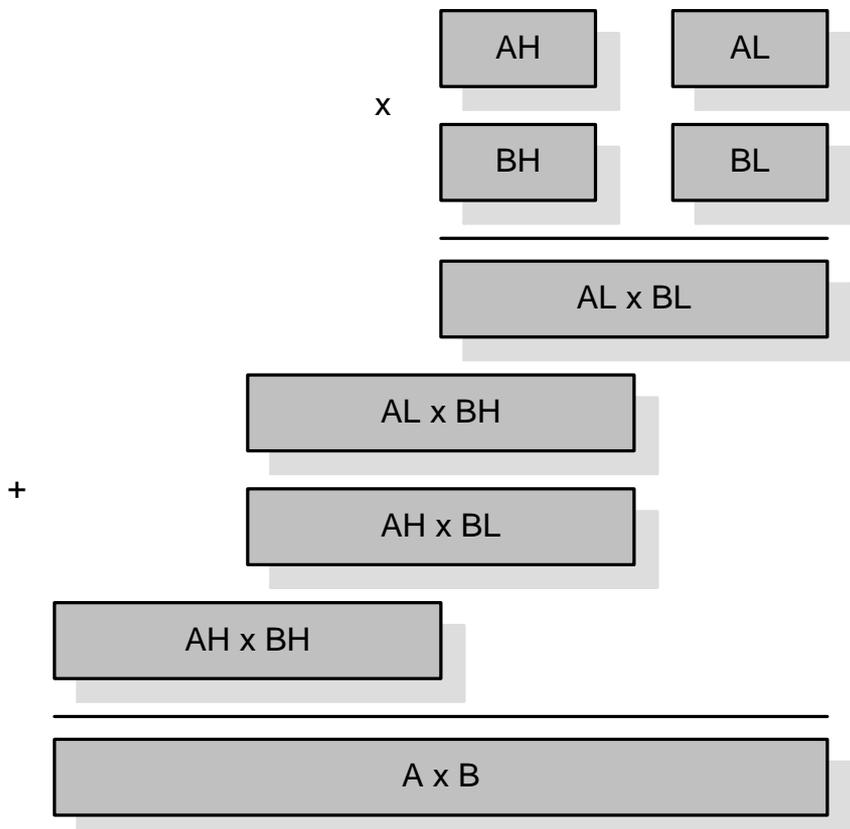
Архитектура ЭВМ

ИУ7



11

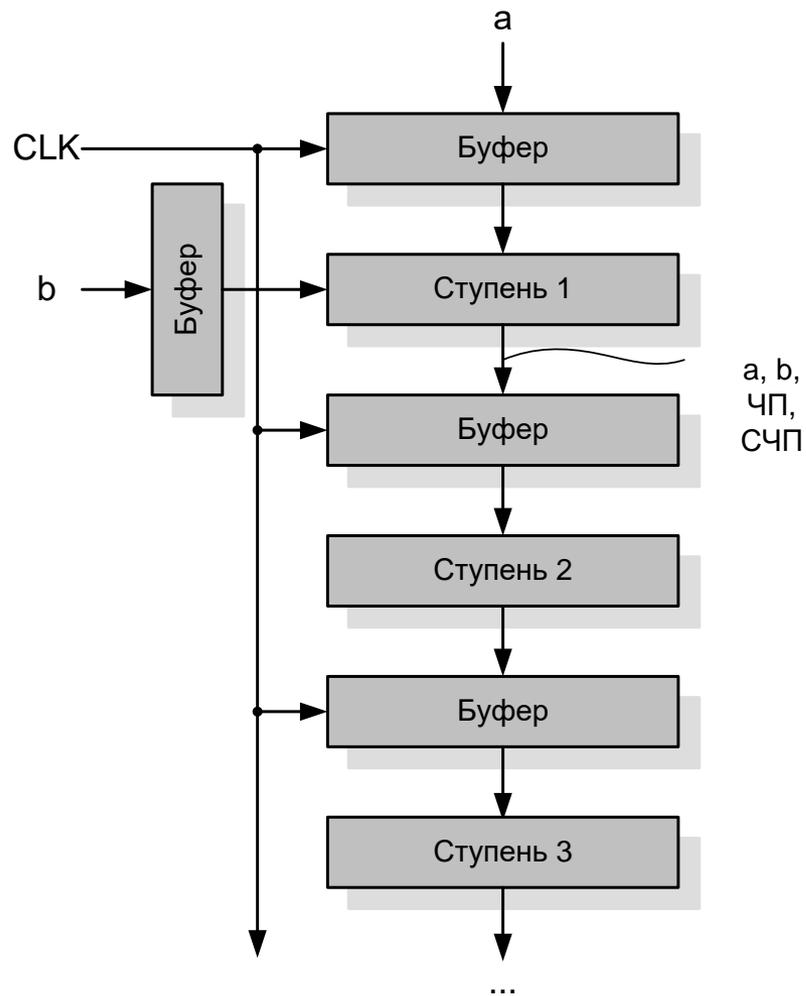
Наращивание размерности умножителей



Архитектура ЭВМ

ИУ7

Конвейеризация умножителей



12

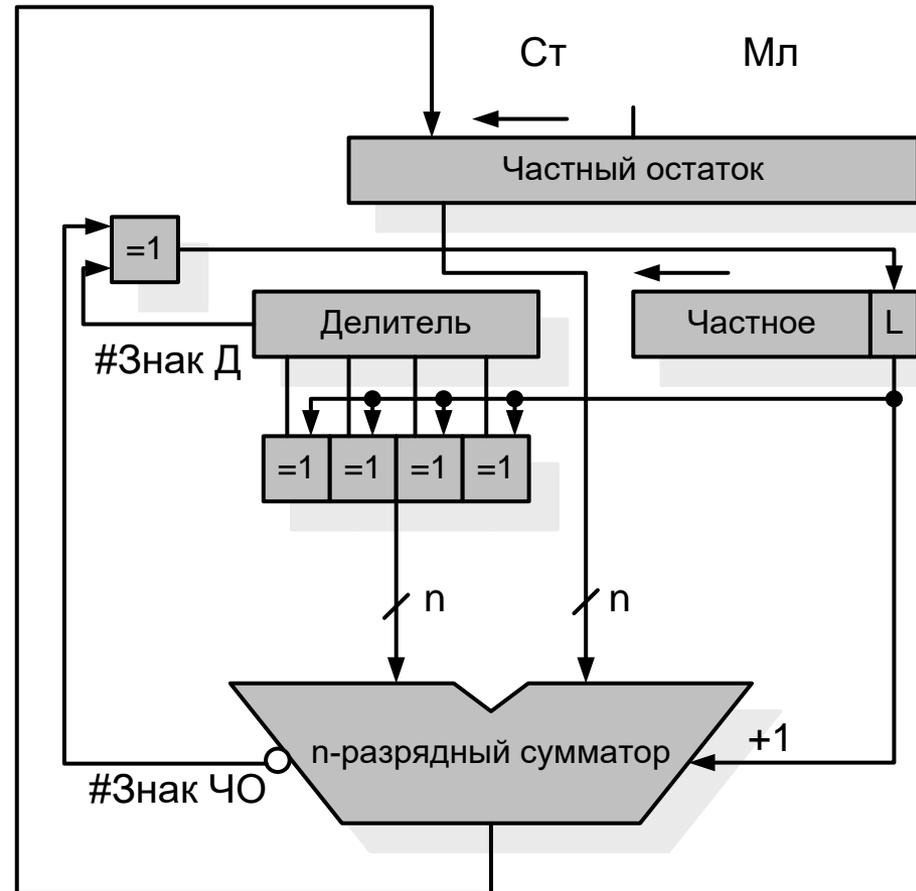
Деление без восстановления остатка

			17										
Делимое			0	1	0	0	0	1	0	1	1	3	
ЧО			0	1	0	0	0	1	1	0	1	5	
<-ЧО	-	0	1	0	0	0	1	0					
-Делитель			0	1	1								
ЧО>0		0	0	0	1	0	1	0					
<-ЧО	-	0	0	0	1	0	1	0	0				
-Делитель			0	1	1								
ЧО<0		1	1	1	1	1	0	0					
<-ЧО	+	1	1	1	1	1	0	0	0				
+Делитель			0	1	1								
ЧО>0		0	0	0	0	0	0	0	0	1	0		

Алгоритм:

- 1) ЧО = Делимое*2;
- 2) ЧО = ЧО – Делитель * 2ⁿ;
- 3) ЧО = ЧО*2;
- 4) Если ЧО<0 то
 - Ч<-0,
 - ЧО = ЧО + Делитель * 2ⁿ
 иначе
 - Ч<-1;
 - ЧО = ЧО – Делитель * 2ⁿ
- 5) Если все цифры то конец
иначе пункт 3.

Схема АЛУ для целочисленного деления



Форматы представления чисел с плавающей запятой (по стандарту IEEE 754 и 784).

Мантисса М числа представляется в нормализованном виде (старший разряд не сохраняется).

$$+ 1.0001 * 2^{+011} = 8.5 = \underbrace{0}_{\text{Знак М}} \underbrace{10000010}_{\text{Порядок + С}} \underbrace{00010\dots0}_{\text{Нормализованная мантисса}}$$

↑
1

Формат	Длина числа	Длина мантиссы	Длина порядка	Смещение порядка	Диапазон чисел
Короткий формат	32	24	8	+127	$10^{-38}..10^{+38}$
Длинный формат	64	53	11	+1023	$10^{-308}..10^{+308}$
Расширенный формат	80	64	15	+16383	$10^{-4932}..10^{+4932}$
Архитектура ЭВМ	ИУ7			15	

Операции над числами с плавающей запятой.

1. Подготовительный этап.

- Разделение упакованного ЧПЗ на группы М,П,З.
- Проверка на специальное числовое значение.

2. Выполнение операции.

- Приведение порядков.
- Определение знака результата.
- Определение мантиссы результата.
- Определение порядка результата.
- Проверка на переполнение, потери значимости мантиссы, потери значимости порядка, неточности, деления на 0.

3. Заключительный этап.

- Проверка на специальное числовое значение.
- Нормализация результата.
- Проверка на переполнение, потери значимости мантиссы, потери значимости порядка, неточности.
- Упаковка полей З,П,М в ЧПЗ.

Организация операций сложения и вычитания над числами с плавающей запятой.

1. Подготовительный этап
2. Определение меньшего из двух порядков и проведение операции выравнивания порядков (сдвиг вправо на разность порядков).
3. Проверка на потерю значимости одного операнда (неточность).
4. Определение результирующего порядка как максимума.
5. Сложение мантисс и определение знака результата.
6. Проверка на переполнение мантиссы. Если да, то сдвигаем мантиссу вправо и увеличиваем порядок на 1.
7. Проверка на переполнение порядка.
8. Заключительный этап.

Организация операций умножения чисел с плавающей запятой.

1. Подготовительный этап
2. Проверка ($M1=0$ или $M2=0$). Если да, то $P=0$.
3. Определение порядка результата: $P_r = P_1 + P_2 - C$.
4. Проверка на переполнение порядка.
5. Определение мантиссы результата: $M_r = M_1 * M_2$.
6. Определение знака результата.
7. Заключительный этап.

Организация операций деления чисел с плавающей запятой.

1. Подготовительный этап
2. Проверка ($M1=0$ или $M2=0$). Если деление на ноль, то +/-бесконечность или ошибка.
3. Определение порядка результата: $P_r = P_1 - P_2 + C$.
4. Проверка на переполнение порядка.
5. Определение мантиссы результата: $M_r = M_1 * (1/M_2)$.
6. Определение знака результата.
7. Заключительный этап.

TABLE 6-5 Floating-Point Multiplication

MULTIPLICATION Instruction FMUL $rs_1, rs_2 [rs_2, rs_1] \rightarrow rd$	Result from the operation includes one or more of the following: <ul style="list-style-type: none"> • Number in f register. See <i>Trap Event</i> on page 132. • Exception bit set. See TABLE 6-12. • Trap occurs. See abbreviations in TABLE 6-12. • Underflow/overflow can occur. 			
	Masked Exception, TEM = 0		Enabled Exception, TEM = 1	
	Destination Register Written (rd)	Flag(s)	Destination Register Written (rd)	Flag(s), Trap
+0, [+0]+Normal]	+0	None set.	+0	None set.
+0, [-0]-Normal]	-0	None set.	-0	None set.
-0, [+0]+Normal]	-0	None set.	-0	None set.
-0, [-0]-Normal]	+0	None set.	+0	None set.
+0, +Infinity	QNaN	Asserts nvc, nva.	No	Asserts nvc. IEEE trap ¹ enabled.
+0, -Infinity	QNaN	Asserts nvc, nva.	No	Asserts nvc. IEEE trap enabled.
-0, +Infinity	QNaN	Asserts nvc, nva.	No	Asserts nvc. IEEE trap enabled.
-0, -Infinity	QNaN	Asserts nvc, nva.	No	Asserts nvc. IEEE trap enabled.
\pm Normal, \pm Normal	Can underflow/ overflow. See 6.5.		Can underflow/ overflow. See 6.5.	
[+Normal]+Infinity], +Infinity	+Infinity	None set.	+Infinity	None set.
[+Normal]+Infinity], -Infinity	-Infinity	None set.	-Infinity	None set.
[-Normal]-Infinity], +Infinity	-Infinity	None set.	-Infinity	None set.
[-Normal]-Infinity], -Infinity	+Infinity	None set.	+Infinity	None set.

1. IEEE trap means *fp_exception_IEEE_754*.

TABLE 6-3 Floating-Point Addition

ADDITION Instruction	Result from the operation includes one or more of the following:			
	<ul style="list-style-type: none"> • Number in f register. See <i>Trap Event</i> on page 132. • Exception bit set. See TABLE 6-12. • Trap occurs. See abbreviations in TABLE 6-12. • Underflow/overflow can occur. 			
	Masked Exception, TEM = 0		Enabled Exception, TEM = 1	
FADD $rs_1, rs_2 [rs_2, rs_1] \rightarrow rd$	Destination Register Written (rd)	Flag(s)	Destination Register Written (rd)	Flag(s), Trap
+0, +0	+0	None set.	+0	None set.
+0, -0	+0 (FSR.RD = 0,1,2) -0 (FSR.RD = 3)	None set.	+0 (FSR.RD = 0,1,2) -0 (FSR.RD = 3)	None set.
-0, -0	-0	None set.	-0	None set.
± 0 , +Normal	+Normal	None set.	+Normal	None set.
± 0 , -Normal	-Normal	None set.	-Normal	None set.
± 0 , +Infinity	+Infinity	None set.	+Infinity	None set.
± 0 , -Infinity	-Infinity	None set.	-Infinity	None set.
\pm Normal, +Infinity	+Infinity	Asserts ofc, ofa, nvc, nva.	No	Asserts ofc, nvc. IEEE trap ¹ enabled.
\pm Normal, -Infinity	-Infinity	Asserts ofc, ofa, nvc, nva.	No	Asserts ofc, nvc. IEEE trap enabled.
+Normal, +Normal	Can overflow. See 6.5.3.		Can overflow. See 6.5.3.	
+Normal, -Normal	\pm Normal		Normal	
-Normal, +Normal	\pm Normal		Normal	
-Normal, -Normal	Can underflow. See 6.5.4.		Can underflow. See 6.5.4.	
+Infinity, +Infinity	+Infinity	None set.	+Infinity	None set.
+Infinity, -Infinity	QNaN	Asserts nvc, nva.	No	Asserts nvc. IEEE trap enabled.
-Infinity, +Infinity	QNaN	Asserts nvc, nva.	No	Asserts nvc. IEEE trap enabled.
-Infinity, -Infinity	-Infinity	None set.	-Infinity	None set.

1. IEEE trap means *fp_exception_IEEE_754*.

TABLE 6-6 Floating-Point Division

DIVISION Instruction rs_1, rs_2 FDIV $rs_1, rs_2 \rightarrow rd$	Result from the operation includes one or more of the following: <ul style="list-style-type: none"> • Number in f register. See <i>Trap Event</i> on page 132. • Exception bit set. See TABLE 6-12. • Trap occurs. See abbreviations in TABLE 6-12. • Underflow/overflow can occur. 			
	Masked Exception, TEM = 0		Enabled Exception, TEM = 1	
	Destination Register Written (rd)	Flag(s)	Destination Register Written (rd)	Flag(s), Trap
$\pm 0, \pm 0$	sign=0, expo=111...111, frac=111...111 (QNaN)	Asserts <i>nvc, nva</i> .	No	Asserts <i>nvc</i> . IEEE trap ¹ enabled.
$\pm 0, \pm \text{Normal}$	± 0	None set.	± 0	None set.
$\pm 0, \pm \text{Infinity}$	± 0	None set.	± 0	None set.
+Normal, +0	+Infinity	Asserts <i>nvc, nva</i> .	No	Asserts <i>dzc, nvc</i> . IEEE trap enabled.
+Normal, -0	-Infinity	Asserts <i>nvc, nva</i> .	No	Asserts <i>dzc, nvc</i> . IEEE trap enabled.
-Normal, +0	-Infinity	Asserts <i>nvc, nva</i> .	No	Asserts <i>dzc, nvc</i> . IEEE trap enabled.
-Normal, -0	+Infinity	Asserts <i>nvc, nva</i> .	No	Asserts <i>dzc, nvc</i> . IEEE trap enabled.
$\pm \text{Normal}, \pm \text{Normal}$	Can underflow/overflow. See 6.5.		Can underflow/overflow. See 6.5.	
$\pm \text{Infinity}, \pm \text{Infinity}$	QNaN	Asserts <i>nvc, nva</i> .	No	Asserts <i>nvc</i> . IEEE trap enabled.
+Infinity, +Normal	+Infinity	None set.	+Infinity	None set.
+Infinity, -Normal	-Infinity	None set.	-Infinity	None set.
-Infinity, +Normal	-Infinity	None set.	-Infinity	None set.
-Infinity, -Normal	+Infinity	None set.	+Infinity	None set.

1. IEEE trap means *fp_exception_IEEE_754*.

Устройства выполнения векторных операций (Эльбрус1, Intel, AMD, Sun, IBM, MIPS).

Устройство выполнения целочисленных MMX операций (MultiMedia eXtensions, Intel) и SSE операций (Streaming SIMD Extension) предназначены для ускорения приложений, ориентированных на выполнение однотипных действий с большими массивами целочисленных и вещественных данных. С данными такого типа обычно работают мультимедийные, графические и коммуникационные программы.

Операнды MMX и SSE операций упакованы в группы по 32,64,80,128 разрядов. Выполнение арифметических операций над операндами группы выполняются параллельно.

Технология SIMD (ITM и BT, Эльбрус 1) 1978 год

Технология MMX (Intel Pentium MMX, Intel P6, ...) 1992 год

Технология SSE (Intel P6, Intel NetBurst, ...)

Технология 3DNow (AMD K6, ...)

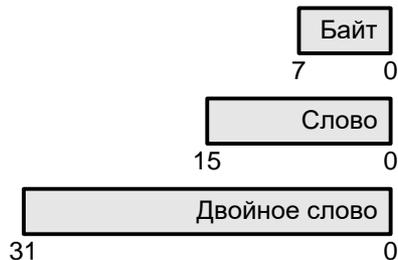
Технология AltiVec (IBM PowerPC)

Технология VIS (Sun UltraSPARC II)

Технология ASE (MIPS 24KE, 74K)

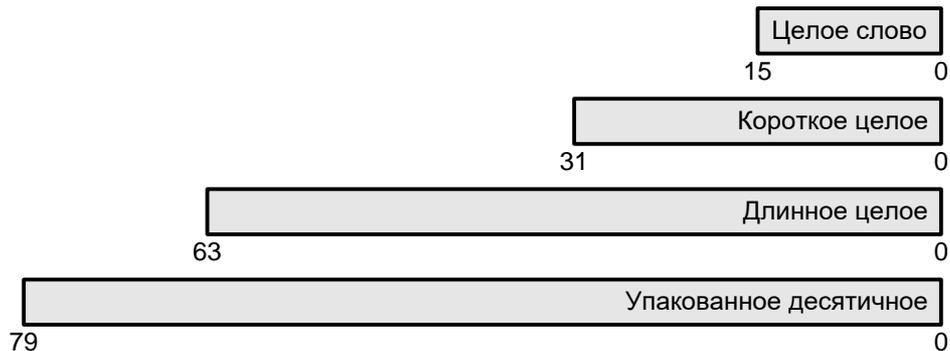
Форматы чисел в микропроцессорах Intel, AMD

Форматы целочисленного АЛУ



Форматы блока обработки ЧПЗ

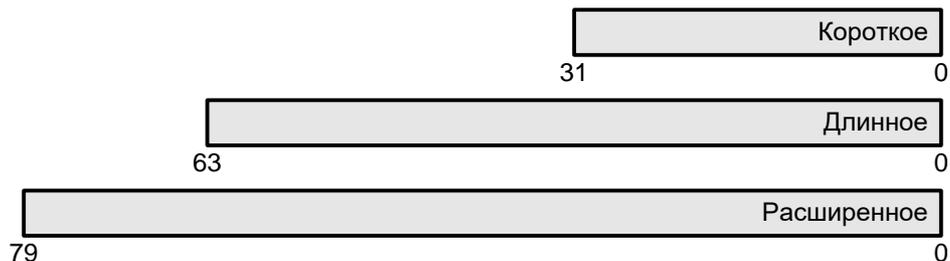
Целочисленные



Форматы MMX



Числа с плавающей запятой



Форматы SSE

