

Тема.

**Арифметико-логические
устройства,
микропроцессоры**



Сумматоры

По числу выводов различают: полусумматоры, одноразрядные сумматоры, многоразрядные сумматоры.

Полусумматор - устройство для сложения двух одноразрядных кодов, имеет два входа и два выхода и формирующее из сигналов входных слагаемых сигналы суммы и переноса в старший разряд.

Однозарядный сумматор – устройство для сложения двух одноразрядных кодов, имеющее три входа и два выхода, и формирующее из сигналов входных слагаемых и сигнала переноса из младших разрядов сигналы суммы и переноса в старший разряд.

Многоразрядный сумматор – устройство для сложения двух многоразрядных кодов, формирующее на выходе код суммы и сигнал переноса в случае, если результат сложения не может быть представлен кодом, разрядность которого совпадает с разрядностью кодов слагаемых.

Многоразрядные сумматоры подразделяются на

последовательные и параллельные. В последовательных сумматорах операция сложения выполняется последовательно разряд за разрядом, начиная с младшего.

Различают комбинационные сумматоры — устройства, не имеющие собственной памяти, и накапливающие сумматоры, снабжённые собственной внутренней памятью, в которой аккумулируются результаты выполненной операции. При этом каждое очередное слагаемое прибавляется к уже имевшемуся в устройстве значению.

По способу тактирования различают синхронные и асинхронные сумматоры. В синхронных сумматорах время выполнения операции арифметического суммирования двух кодов не зависит от вида самих кодов и всегда остается постоянным. В асинхронных сумматорах время выполнения операции зависит от вида слагаемых. Поэтому по завершении выполнения суммирования необходимо вырабатывать специальный сигнал завершения операции.

В зависимости от используемой системы счисления различают двоичные, двоично-десятичные и другие типы сумматоров.

Сложение
одноразрядных
двоичных кодов -
операция "Ис
ключающее ИЛИ"
над исходным
слагаемыми x_1 и x_0 .

Сумматор - один из основных узлов арифметико-логического устройства. Термин сумматор охватывает широкий спектр устройств от простейших логических схем, до сложнейших цифровых узлов. Общим для этих устройств является арифметическое сложение двоичных чисел

**Таблица истинности сложения
двух одноразрядных
двоичных кодов**

x_1	x_0	s	p
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$s = \bar{x}_1 x_0 + x_1 \bar{x}_0 = x_1 \oplus x_0$$

$$p = x_1 x_0.$$

$$\begin{aligned} \bar{s} &= \overline{x_1 \oplus x_0} = \overline{\bar{x}_1 x_0 + x_1 \bar{x}_0} = \overline{(\bar{x}_1 x_0) (x_1 \bar{x}_0)} = (\bar{x}_1 + x_0) (\bar{x}_1 + x_0) = \\ &= x_1 \bar{x}_1 + x_1 x_0 + \bar{x}_1 \bar{x}_0 + \bar{x}_0 x_0 = \bar{x}_1 \bar{x}_0 + x_1 x_0. \end{aligned}$$

**Таблица истинности сложений
разрядов многоразрядных двоичных
кодов**

x_1	x_0	P_{-1}	s	P
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	1
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

$$s = (x_1 \oplus x_0) \bar{P} + \overline{(x_1 \oplus x_0)} P_{-1},$$

$$P = x_1 x_0 + (x_1 \oplus x_0) P_{-1}.$$

Сложение многоразрядных двоичных кодов.

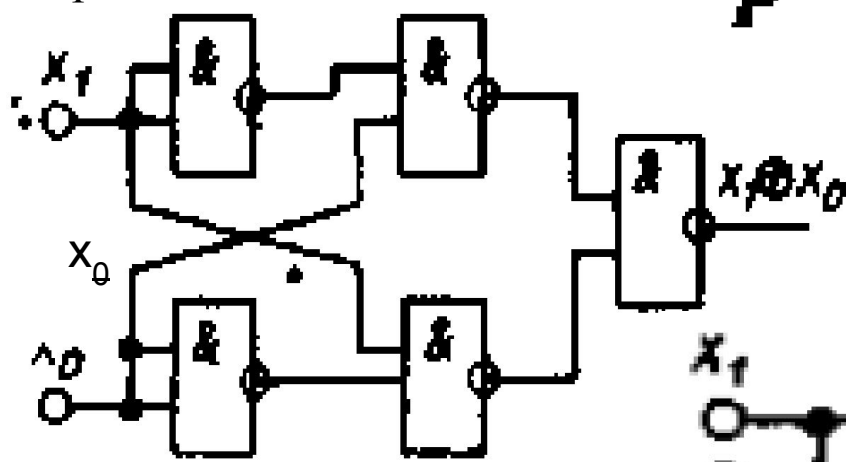
Для получения суммы двух старших разрядов необходимо сначала выполнить операцию Исключающее ИЛИ над исходным слагаемыми x_1 и x_0 и затем еще одну операцию Исключающее ИЛИ над результатом первой операции Исключающее ИЛИ и сигналом переноса из предыдущего разряда. Для получения сигнала переноса также необходимо воспользоваться результатом операции Исключающее ИЛИ над слагаемыми x_1 и x_0 .

ДВОИЧНЫЙ ПОЛУСУММАТОР - устройство

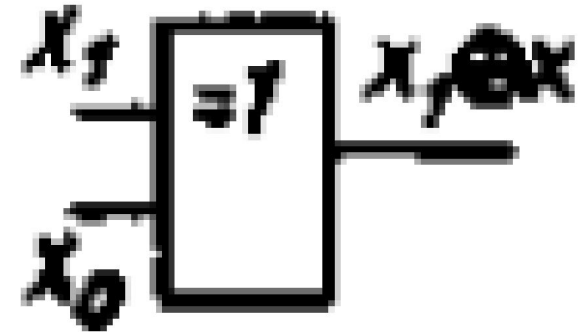
сложения двух одноразрядных кодов, имеет два входа и два выхода, формирует из сигналов входных слагаемых сигналы суммы и переноса в старший разряд.

$$S = x_1 \oplus x_0 = \bar{x}_1 x_0 + x_1 \bar{x}_0 = (\bar{x}_1 | x_0) | (x_1 | \bar{x}_0)$$

Структурная схема реализации операции «Исключающее ИЛИ»



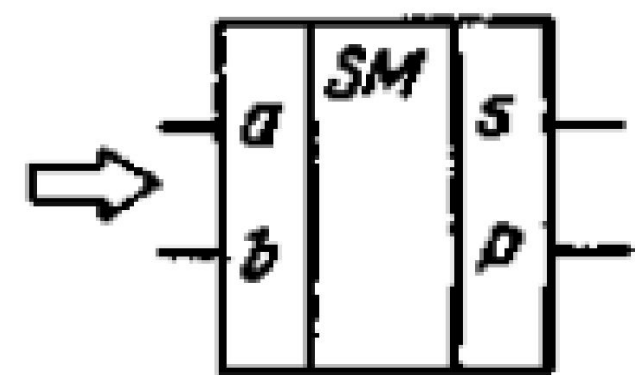
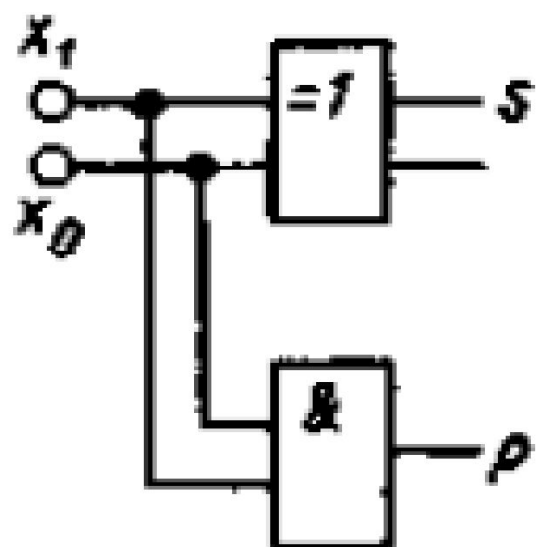
$$P = x_1 x_0$$



Структурная схема реализации двоичного полусумматора

Время операции

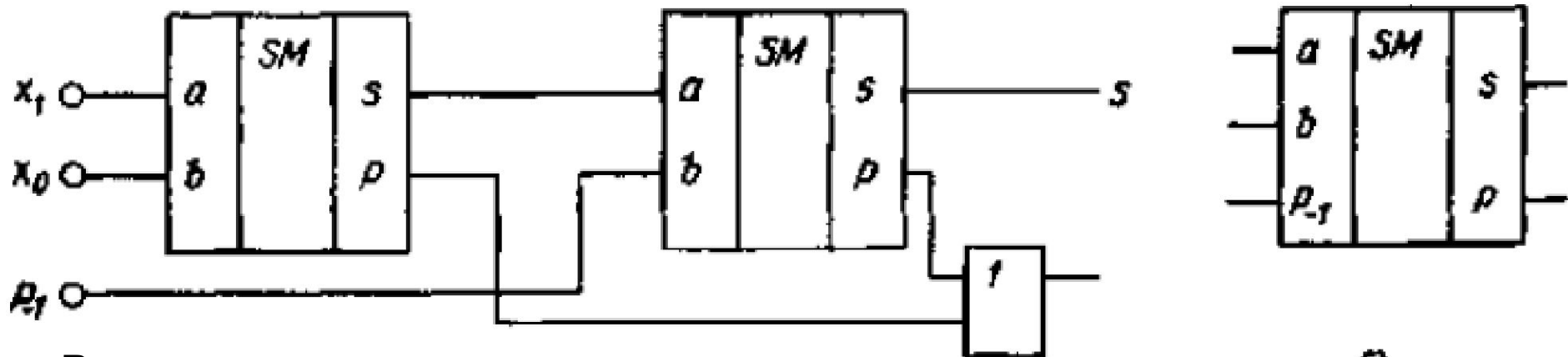
$$t_{п \Sigma} = 3t_{\text{лп}}$$



ОДНОРАЗРЯДНЫЙ СУММАТОР - устройство, предназначенное для сложения двух одноразрядных кодов, имеющее три входа и два выхода, и формирующее из сигналов входных слагаемых и сигнала переноса из младших разрядов сигналы суммы и переноса в старший разряд.

$$s = (x_1 \oplus x_0) \bar{p}_1 \mid \overline{(x_1 \oplus x_0) p_{-1}},$$

$$p = x_1 x_0 \mid (x_1 \oplus x_0) p_{-1}.$$



Время операции

$$t_{0 \Sigma} = 2t_{п \Sigma} = 6t_{з \rho}$$

$$t_{сн} = 5t_{з \rho}$$

Время суммирования и приведённой схеме также определяется временем выполнения операции Исключающее ИЛИ

Формирование сигнала переноса в старший разряд выполняется быстрее

МНОГОРАЗРЯДНЫЙ СУММАТОР - устройство сложения двух многоразрядных кодов, формирующее на выходе код суммы и сигнал переноса в случае, если результат сложения не может быть представлен кодом, разрядность которого совпадает с разрядностью кодов слагаемых. Многоразрядные сумматоры подразделяются на последовательные и параллельные.

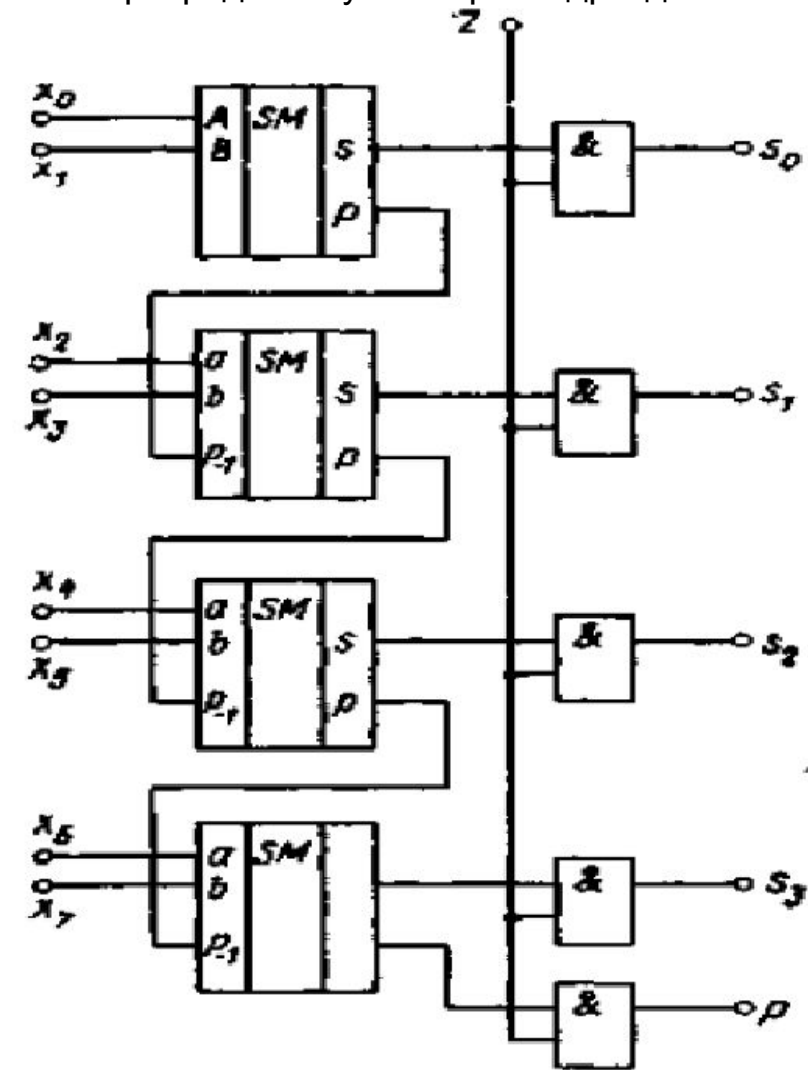


Рис. 18.5. Структурная схема параллельного многоразрядного сумматора с последовательным переносом

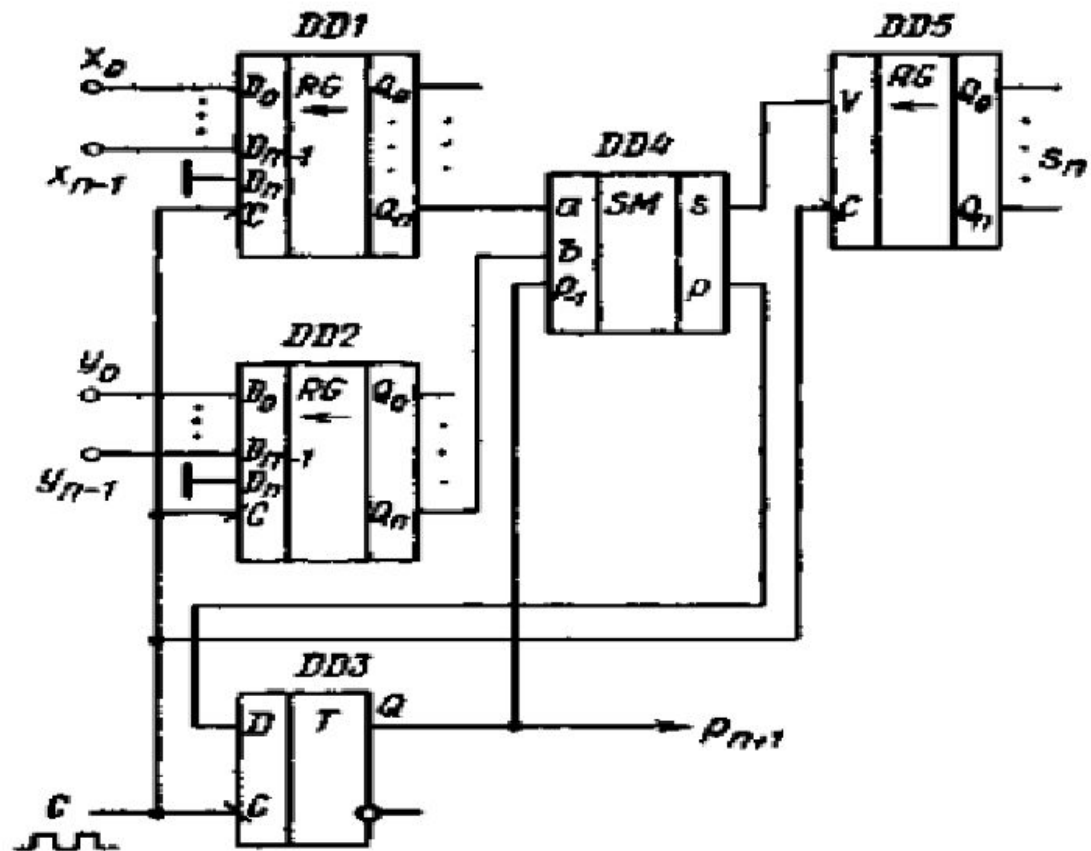


Рис. 18.6. Структурная схема многоразрядного сумматора последовательного действия

Двоично-десятичный код

Двоичный код $x_3x_2x_1x_0$	Двоично десятичный код	Десятичное число
1	2	3
0 0 0 0	0 0 0 0	0
0 0 0 1	0 0 0 1	1
0 0 1 0	0 0 1 0	2
0 0 1 1	0 0 1 1	3
0 1 0 0	0 1 0 0	4
0 1 0 1	0 1 0 1	5
0 1 1 0	0 1 1 0	6
0 1 1 1	0 1 1 1	7
1 0 0 0	1 0 0 0	8
1 0 0 1	1 0 0 1	9
1 0 1 0	0 0 0 1	10
1 0 1 1	0 0 0 1	11
1 1 0 0	0 0 0 1	12
1 1 0 1	0 0 0 1	13
1 1 1 0	0 0 0 1	14
1 1 1 1	0 0 0 1	15

$$F = x_3(x_1 + x_2) + P.$$

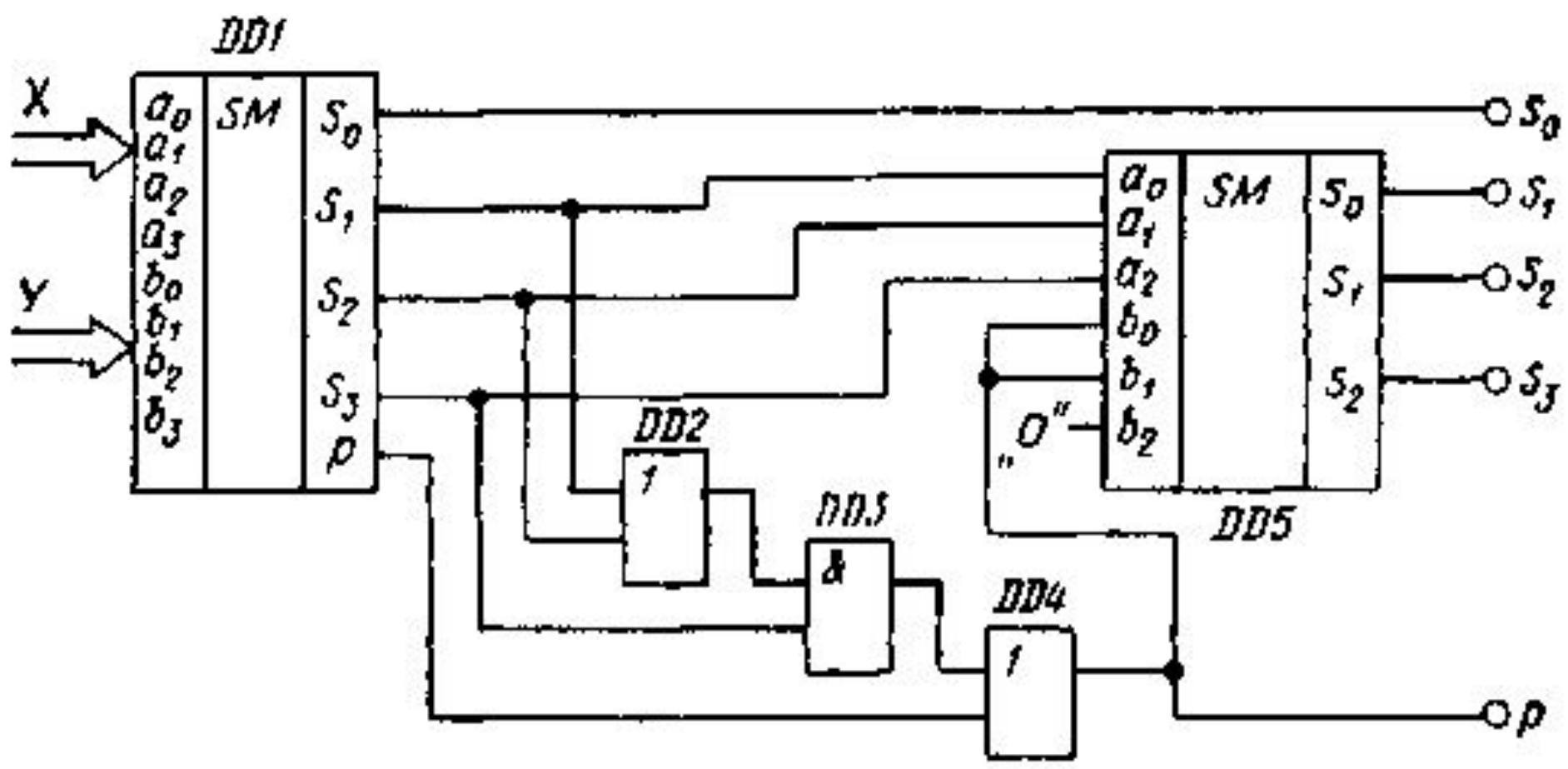


Рис 18 12 Двоинно десятичный сумматор

$$0-0=0$$

$$0-1=\overset{\cdot}{1}1$$

$$1-0=1$$

$$1-1=0$$

$$\begin{array}{r} \overset{\cdot}{1} \overset{\cdot}{1} 0 \\ - \quad 11 \\ \hline 011 \end{array}$$

$$\begin{array}{r} \overset{\cdot}{1} \overset{\cdot}{1} \overset{1}{0} \overset{1}{0} 0 1 \\ - \quad 11111 \\ \hline 10010 \end{array}$$

АЛГОРИТМ ВЫЧИТАНИЯ ДВОИЧНЫХ ЧИСЕЛ

Известно, что операцию вычитания двух чисел можно заменить операцией сложения, если в качестве вычитаемого взять число, противоположное по знаку исходному. Поэтому для выполнения операции вычитания можно использовать схемы сумматоров.

Для представления отрицательных чисел наибольшее распространение получил метод, в котором для обозначения знака используется старший разряд его двоичного кода.

Так, если используется 8-разрядный двоичный код, то семь его разрядов содержат информацию о величине (модуле) числа, а восьмой (старший) — о его знаке: присутствие в этом разряде нуля означает положительное число, присутствие единицы — отрицательное. Используя 8-разрядный код, можно записать числа с десятичными эквивалентами от -127 до $+127$.

Присутствие знакового разряда непосредственно не позволяет использовать рассмотренные схемы сумматоров для выполнения операции вычитания чисел, представленных в прямом двоичном коде. Наиболее часто для записи отрицательного числа используется дополнительный код.

**Алгоритм получения дополнительного кода двоичного числа :
- записывают обратный код исходного числа (все его разряды инвертируют -заменяют дополнениями);**

- к полученному после инвертирования коду добавляют единицу.

Далее, при сложении полученного таким образом кода вычитаемого с кодом уменьшаемого будет реализована операция вычитания. Знак результата определяет старший разряд полученного кода. Если старший разряд равен нулю, получено положительное число, представленное в прямом коде. Если старший разряд равен единице, получено отрицательное число, представленное в дополнительном коде.

Пример 8.1. Используя дополнительный код, найти разность $31-12$.

Решение. 1 Прямые двоичные коды заданных чисел

$$31_{10} = 00011111_2; 12_{10} = 00001100_2.$$

2 Обратный код вычитаемого 1110011

3 Дополнительный код вычитаемого $1110011 + 00000001 = 11110100$.

4. Сложение

$$\begin{array}{r} 00011111 \\ + \\ 11110100 \\ \hline 100010011 \end{array}$$

Полученный результат представлен 9-разрядным двоичным кодом. В этом случае самый старший девятый разряд отбрасывают. Полученное двоичное число содержит нуль в старшем разряде. Поэтому результат положителен и представлен в прямом коде. Нетрудно видеть, что его десятичный эквивалент равен 19.

Пример 18.2. Используя дополнительный код, найти разность $12 - 31$.

Решение. 1. Обратный код вычитаемого 11100000 .

2. Дополнительный код вычитаемого $11100000 + 00000001 = 11100001$

3. Сложение

$$\begin{array}{r} 00001100 \\ + \\ 11100001 \\ \hline 11101101 \end{array}$$

Старший разряд полученного результата равен единице. Следовательно, получено отрицательное число, записанное в дополнительном коде. Для получения прямого кода можно воспользоваться приведенным выше алгоритмом преобразования прямого кода в дополнительный. Однако существует более простое правило такого преобразования: дополнительный код просматривают справа налево, оставляя без изменения все встретившиеся при этом значения лог. 0. Первую встретившуюся лог. 1 также пропускают, а далее все цифры инверсируют. Результатом такого преобразования будет прямой код числа.

Проделав описанные действия с полученным кодом, получим код 00010011 , что с учетом знака эквивалентно числу -19_{10} .

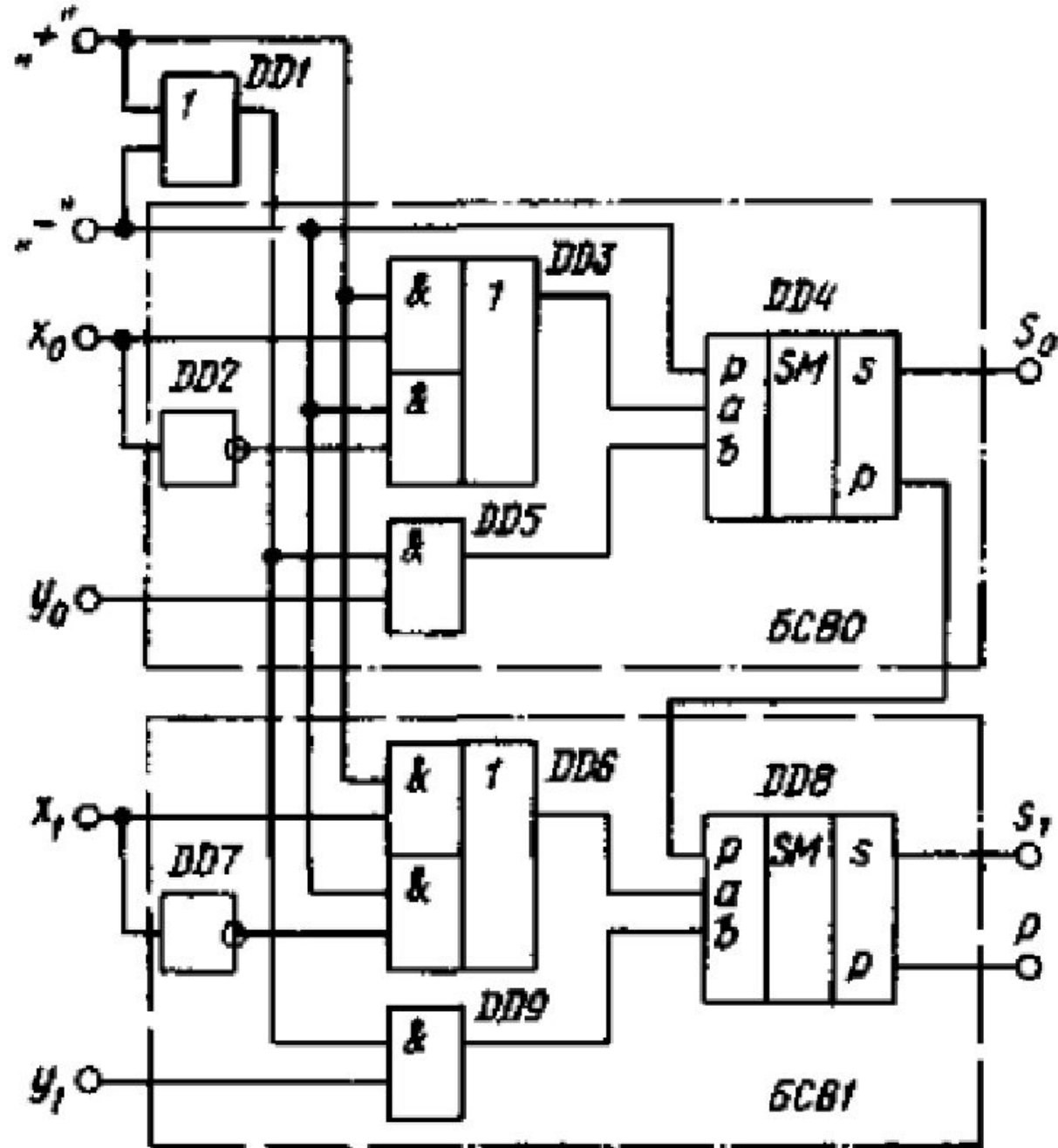


Рис. 1811. Структурная схема сложения-вычитания 2-разрядных кодов

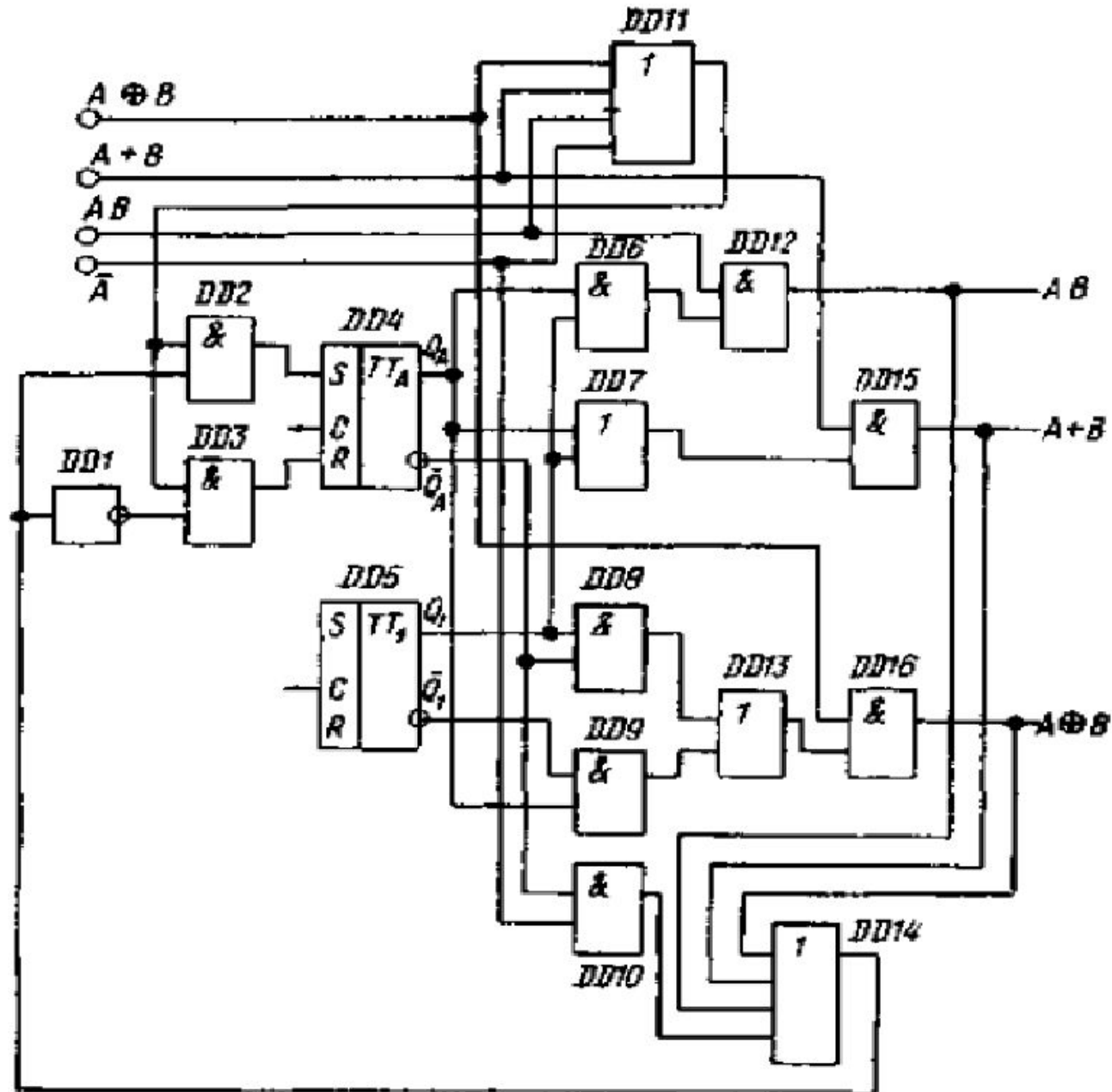
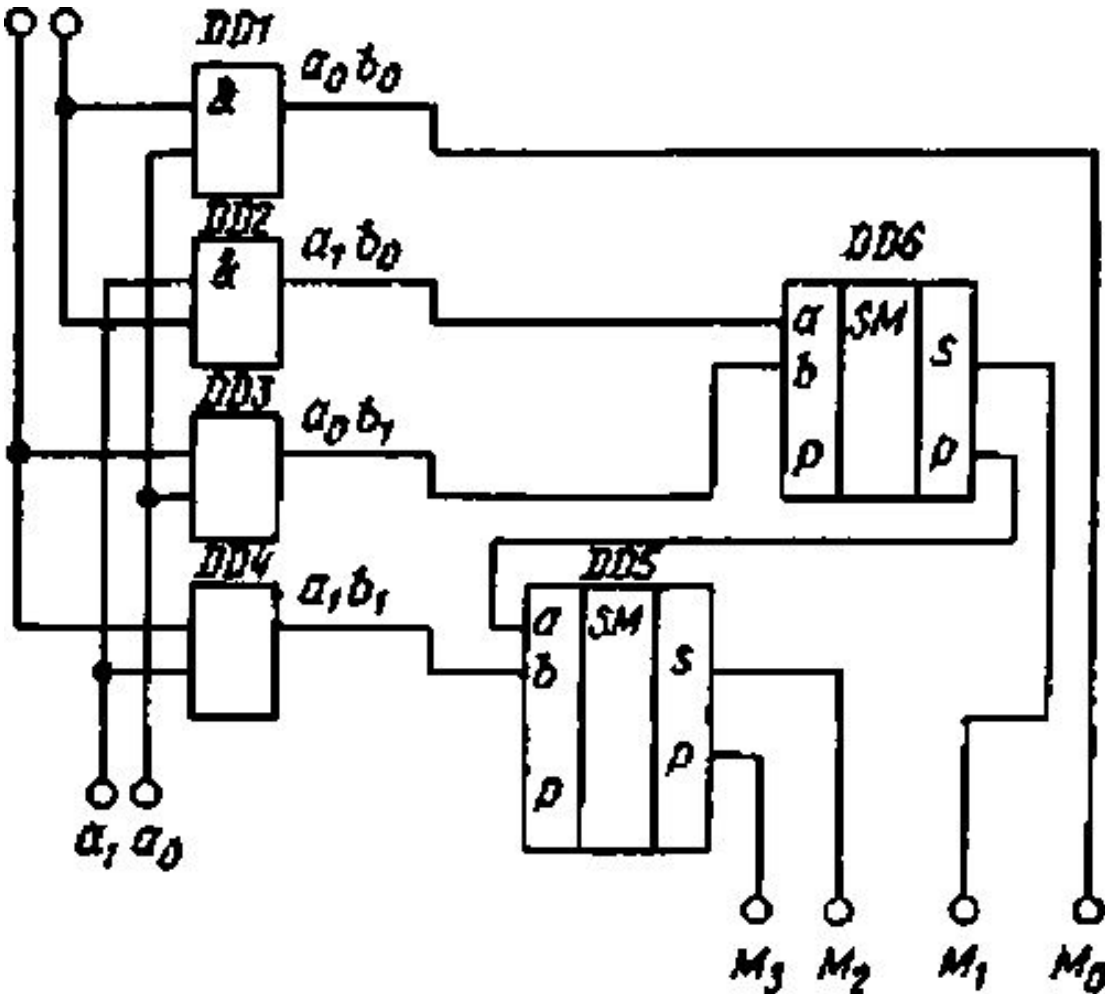


Рис. 18.13. Структурная схема реализации операций логического сложения, вычитания, суммы по модулю два и инверсии

Логика построения аппаратных умножителей неразрывно связана с традиционным алгоритмом выполнения операции умножения, базирующемся на суммировании частных произведений разрядов сомножителей.

$$\begin{array}{r}
 \times \begin{array}{cc} a_1 & a_0 \\ b_1 & b_0 \end{array} \\
 \hline
 + \begin{array}{ccc} & b_1 a_1 & b_1 a_0 \\ & & b_0 a_1 & b_0 a_0 \end{array} \\
 \hline
 \begin{array}{cccc} M_3 & M_2 & M_1 & M_0 \end{array}
 \end{array}$$



Операция умножения сводится к сдвигу числа и сложению, эти операции последовательно выполняет АЛУ (в десятичном счислении $11 * 5 = 55$).

$$\begin{array}{r}
 \times 1011 \\
 \quad 101 \\
 \hline
 \quad 1011 \\
 + 1011 \\
 \hline
 110111
 \end{array}$$

$$\begin{array}{cccc}
 a_3 & a_2 & a_1 & a_0 \\
 \times & & & \\
 b_3 & b_2 & b_1 & b_0
 \end{array}$$

$$\begin{array}{cccc}
 \boxed{b_3 a_3} & \boxed{b_3 a_2} & \boxed{b_3 a_1} & \boxed{b_3 a_0} & \text{Блок 2} \\
 \text{Блок 4} & \boxed{b_2 a_3} & \boxed{b_2 a_2} & \boxed{b_2 a_1} & \boxed{b_2 a_0}
 \end{array}$$

+

$$\begin{array}{cccc}
 \boxed{b_1 a_3} & \boxed{b_1 a_2} & \boxed{b_1 a_1} & \boxed{b_1 a_0} & \text{Блок 1} \\
 \text{Блок 3} & \boxed{b_0 a_3} & \boxed{b_0 a_2} & \boxed{b_0 a_1} & \boxed{b_0 a_0}
 \end{array}$$

$$M_7 \quad M_6 \quad M_5 \quad M_4 \quad M_3 \quad M_2 \quad M_1 \quad M_0$$

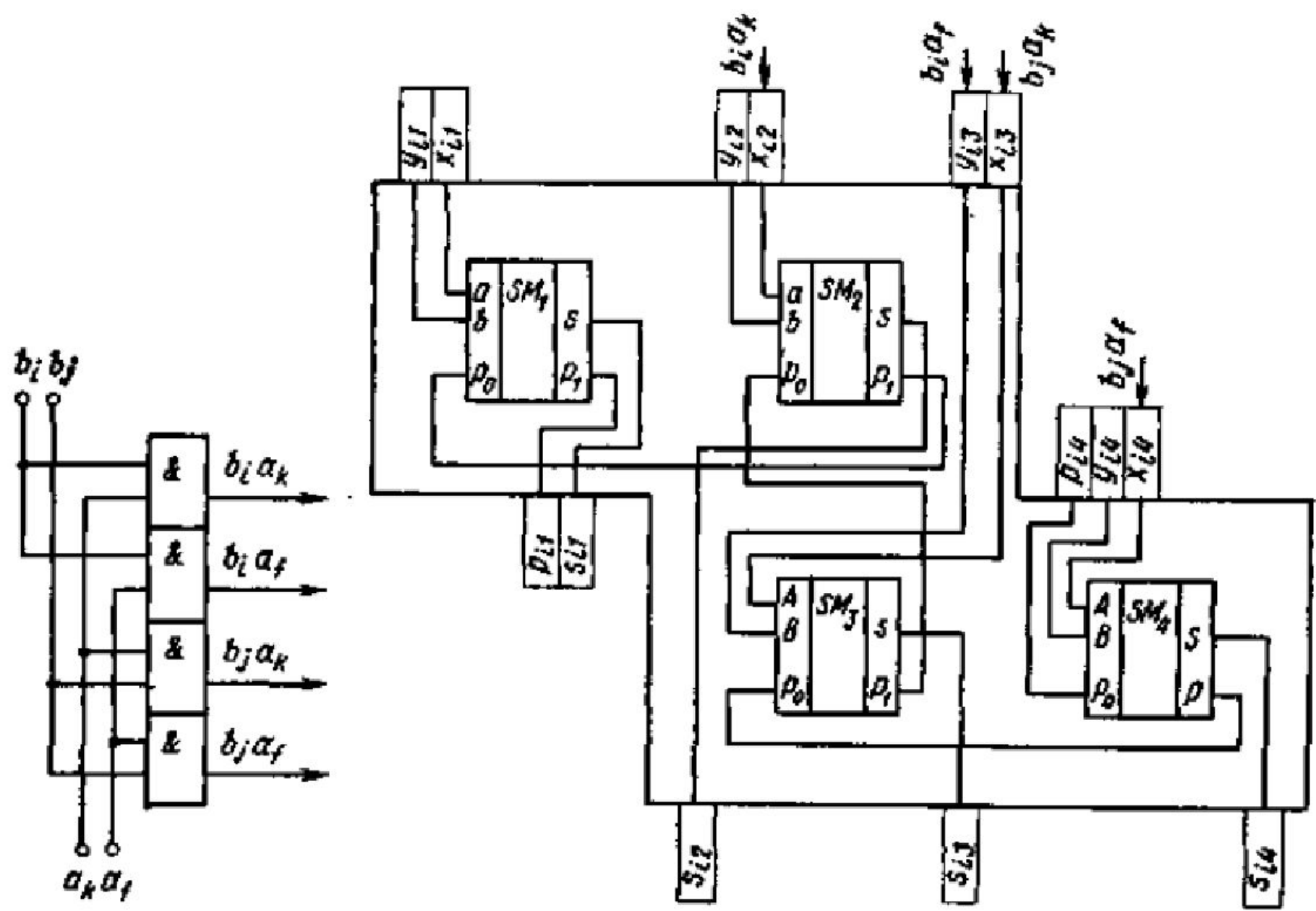


Рис. 18.16. Структурная схема матричного умножителя двоичных кодов, допускающая увеличение разрядности сомножителей

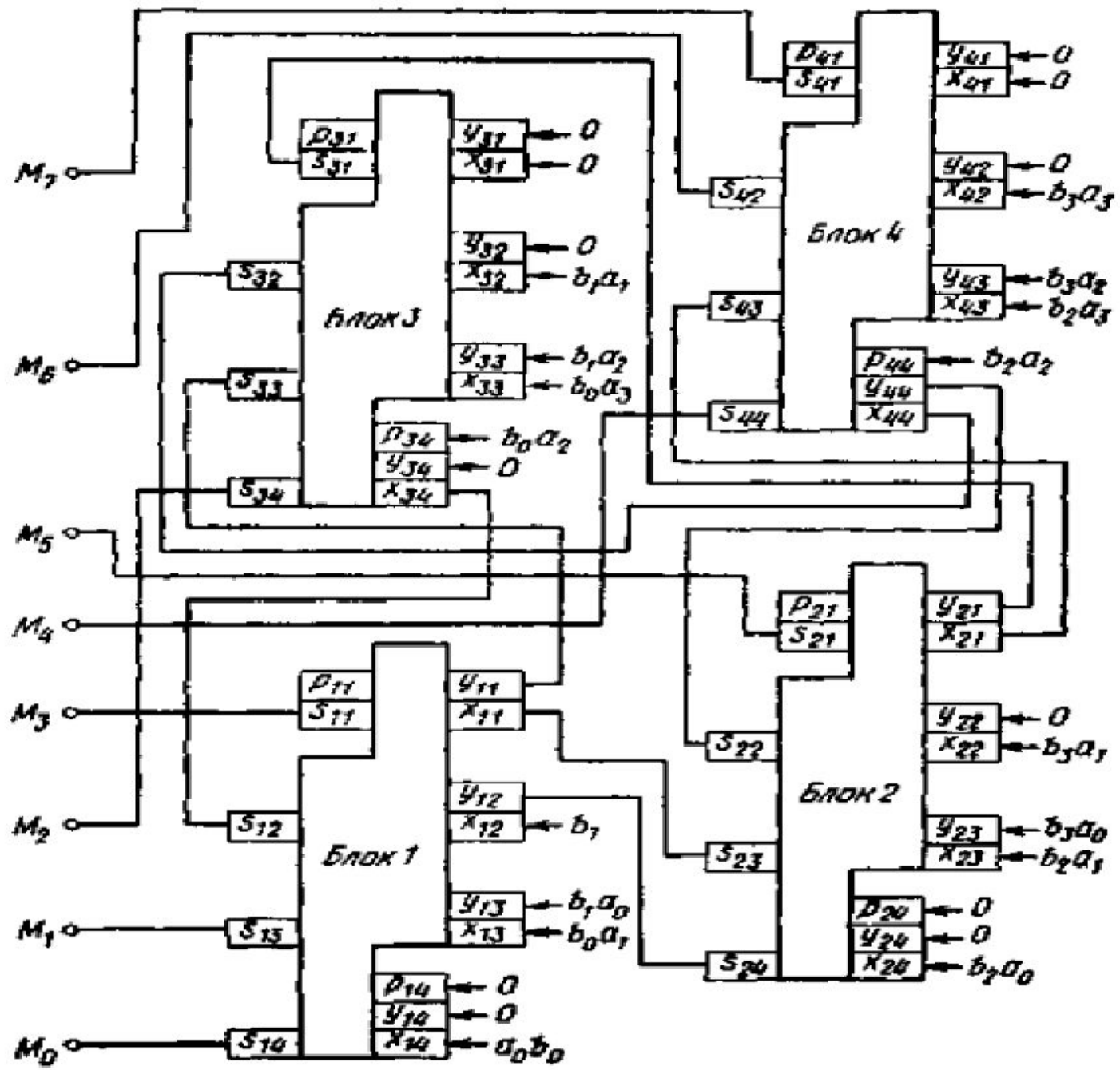


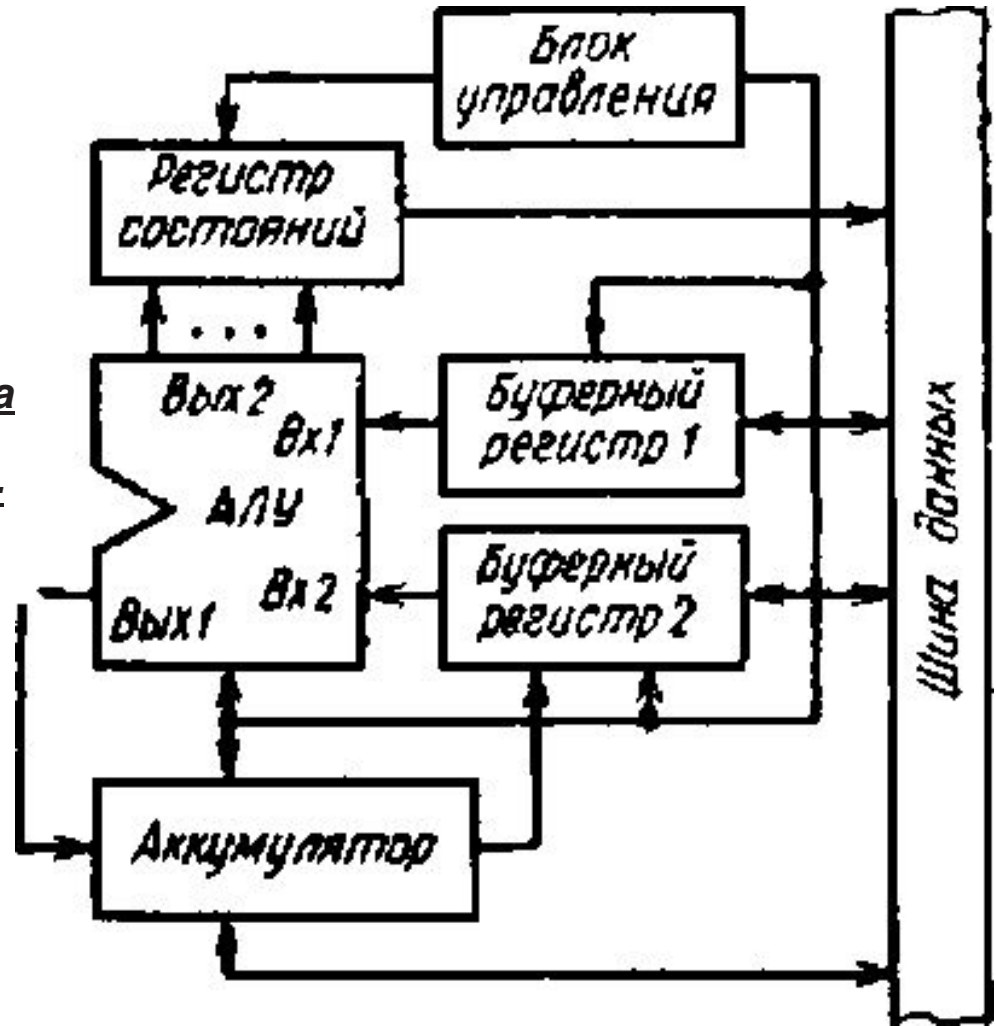
Рис. 18.17. Структурная схема матричного умножителя 4-разрядных кодов

Арифметико-логическое устройство (АЛУ) — часть центрального процессора, формирующая функции двух входных переменных и порождающая одну выходную переменную. Эти функции состоят из простых арифметических и (или) логических операций, операций сдвига.

Вне зависимости от того, насколько широк круг операций, реализуемых современными АЛУ, главными среди них остаются операции арифметического сложения и умножения, продолжительность выполнения которых указывается в качестве основных характеристик вычислительного устройства.

Простейшие операции: арифметическое сложение (вычитание), логическое умножение, логическое сложение, сумма по модулю два, инверсия, сдвиг влево, сдвиг вправо, инкремент (положительное приращение), декремент (отрицательное приращение) выполняются в АЛУ с помощью аппаратных средств (схем на логических элементах).

Операция умножения (деления), как правило, выполняется программным способом (с применением последовательного исполнения нескольких операций сложения и сдвига), выполненных аппаратным способом.



По способу действия над операндами АЛУ подразделяют на параллельного и последовательного действия.

По способу представления чисел различают АЛУ для чисел с фиксированной и плавающей запятой (точкой), для десятичных чисел.

По способу организации работы АЛУ делят на синхронные и асинхронные

По характеру используемых элементов и узлов: универсальные, в которых операции для всех форм представления чисел выполняются одними и теми же схемами, и функциональные, в которых операции над операндами с фиксированной и плавающей запятой, с десятичными переменными и логическими переменными осуществляются в отдельных операционных блоках.

По связи с оперативной памятью (ОЗУ) вычислительного устройства различают АЛУ с непосредственной и магистральной связями.

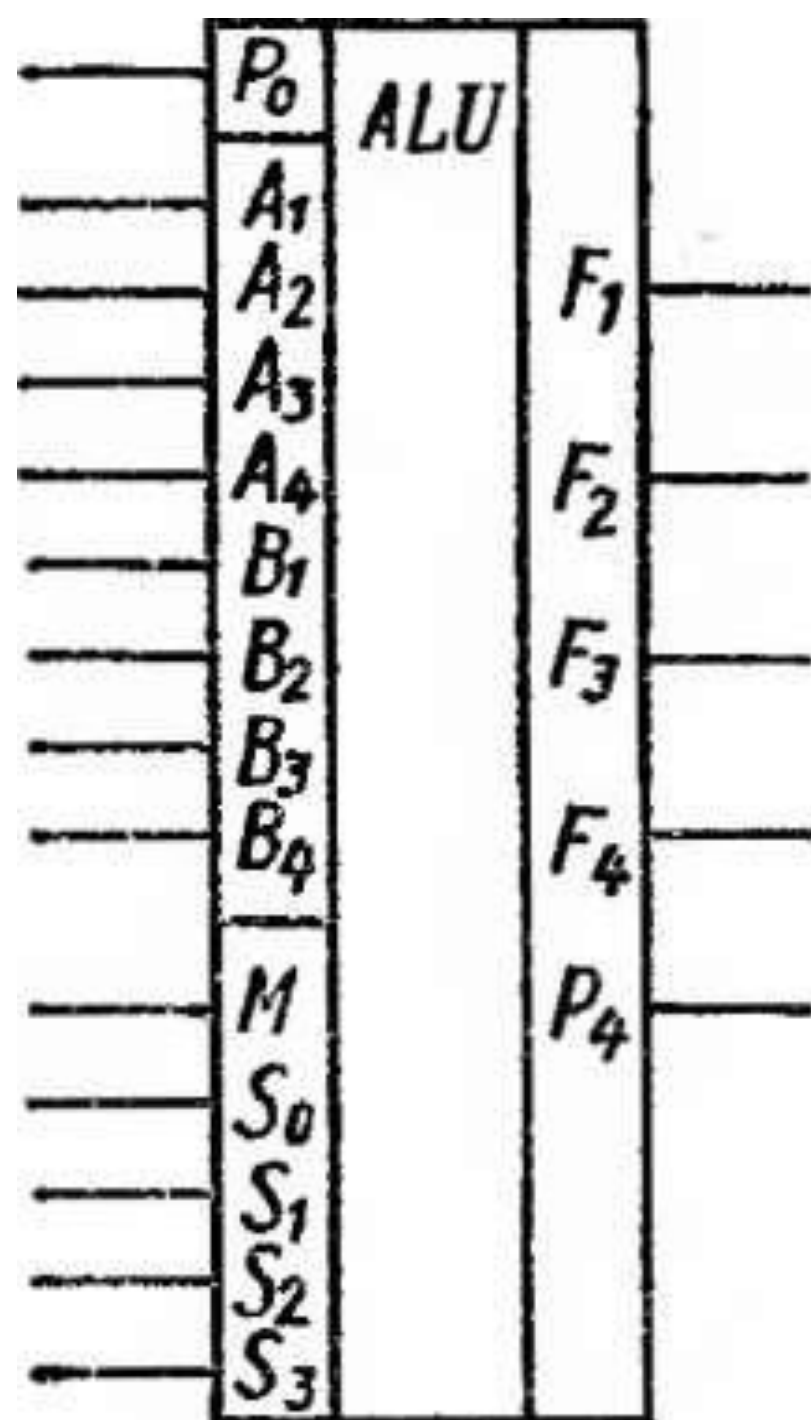
В АЛУ с непосредственной связью схемы управления передачей информации соединены непосредственно с выходами регистров, а в АЛУ с магистральной связью схемы для преобразования информации выделены в отдельные блоки с двумя шинами (входной и выходной), а регистры служат лишь для хранения операндов во время их обработки. АЛУ с магистральной связью используется в процессорах с микропрограммным управлением.

Современная технология изготовления ИМС позволяет выполнить весьма сложные электронные устройства в виде одной или нескольких ИМС. Такое техническое решение будет экономически выгодным, если данное устройство будет производиться массовыми сериями, но, как правило, сложные специализированные устройства не требуется производить в большом числе экземпляров. Это противоречие между возможностями технологии и узкой специализированностью сложных электронных устройств снимается при создании **программируемых** цифровых и логических устройств.

Подобные устройства основаны на работе арифметико-логического устройства (АЛУ), которое выполняется как самостоятельная БИС либо входит в состав других более сложных ИМС. АЛУ позволяет производить арифметические и логические операции над n -разрядными входными кодами (4, 8 или 16); входные коды обозначены A ($A_1 A_2 A_3 A_4$) и B ($B_1 B_2 B_3 B_4$). Сигналы, подаваемые на управляющие входы M, S_0, S_1, S_2, S_3 , определяют, какая именно операция может быть выполнена над входными величинами. Как и в многоразрядном сумматоре, при выполнении арифметических операций на АЛУ может быть подан сигнал переноса из внешней цепи ($P_0=1$), который добавляется в младший разряд АЛУ, на выходе АЛУ формируется сигнал I_4 , переноса из старшего разряда (сигнал переполнения). Сигнал переноса P_{o-1} увеличивает результат (число F) на единицу. АЛУ представляет собой логическую схему, имеющую определённое количество входов ($Lg, Vi, M, S,-$ и P_0) и выходов (Fi, Pn), где Fi — результаты операций в каждом разряде, Pn — сигнал переноса из старшего разряда. Совокупность сигналов на всех входах однозначно определяет совокупность выходных сигналов, поэтому АЛУ является комбинационным устройством.

При управляющем сигнале $\underline{M=0}$ АЛУ выполняет арифметические операции над входными n -разрядными двоичными числами A и B : сложение A и B , сложение A и B с добавлением единицы в младший разряд из внешней цепи (сигнал $P_0 = 1$), вычитание B из A , вычитание B из A с учётом заема единицы из младшего разряда, увеличение или уменьшение числа A на единицу, пересылка чисел A и B с входа АЛУ на выход, сдвиг чисел на один разряд влево (эквивалентно умножению на два) или вправо (эквивалентно делению на два) и др. Комбинация сигналов $S_0 \dots S_3$ определяет, какая именно математическая операция производится АЛУ.

При $\underline{M=1}$ АЛУ выполняет логические операции над функциями A и B (во всех разрядах выполняется одна и та же логическая операция).

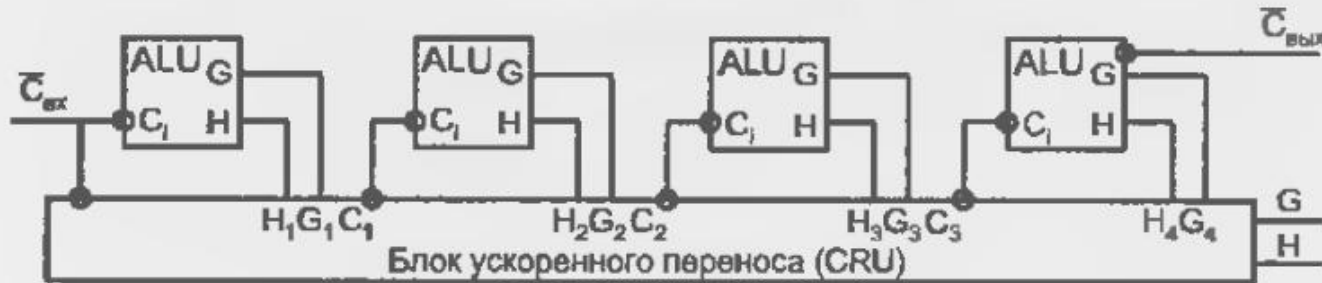


Выбор функции $V_3 V_2 V_1 V_0$	Положительная логика		
	$M = 1$, логические операции	$M = 0$, арифметико-логические операции	
		$\overline{F_{ax}} = 1$	$\overline{F_{ax}} = 0$
0 0 0 0	\overline{A}	A	$A + 1$
0 0 0 1	$\overline{A \vee B}$	$A \vee B$	$A \vee B + 1$
0 0 1 0	\overline{AB}	$A \vee \overline{B}$	$A \vee \overline{B} + 1$
0 0 1 1	0	Минус 1 (дополнение до 2)	0
0 1 0 0	\overline{AB}	$A + \overline{AB}$	$A + \overline{AB} + 1$
0 1 0 1	\overline{B}	$AB + \overline{AB}$	$AB + \overline{AB} + 1$
0 1 1 0	$A \oplus B$	$A - B - 1$	$A - B$
0 1 1 1	$A\overline{B}$	$A\overline{B} - 1$	$A\overline{B}$
1 0 0 0	$\overline{A} \vee B$	$A + AB$	$A + AB + 1$
1 0 0 1	$\overline{A \oplus B}$	$A + B$	$A + B + 1$
1 0 1 0	B	$A \vee \overline{B} + AB$	$A \vee \overline{B} + AB + 1$
1 0 1 1	AB	$AB - 1$	AB
1 1 0 0	1	$A + A^*$	$A + A + 1$
1 1 0 1	$A \vee \overline{B}$	$A \vee B + 1$	$A \vee B + A + 1$
1 1 1 0	$A \vee B$	$A \vee \overline{B} + A$	$A \vee \overline{B} + A + 1$
1 1 1 1	A	$A - 1$	A

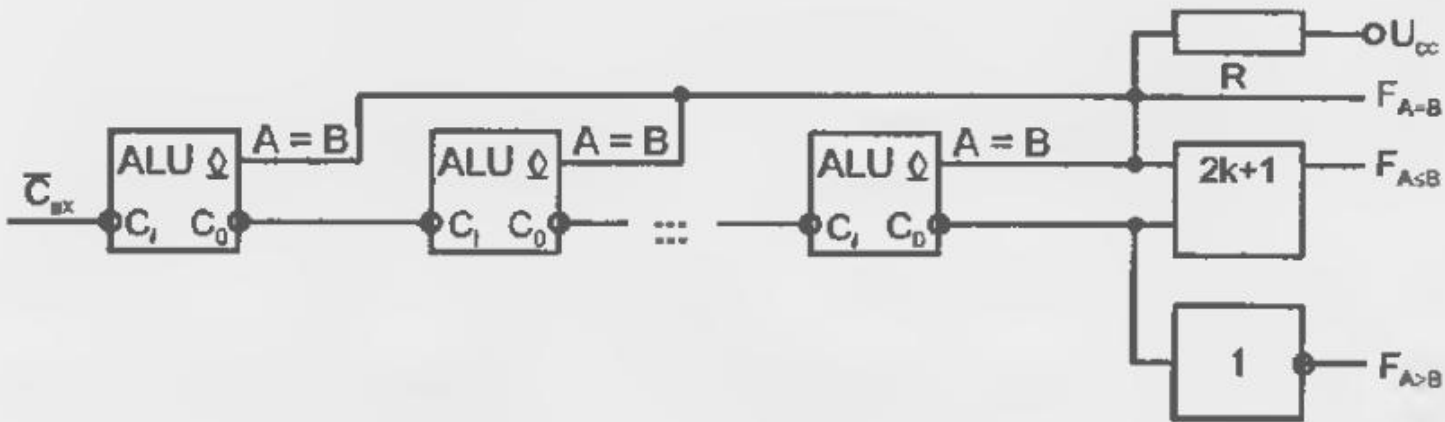
* сдвиг на один разряд влево; \vee – логическое сложение; $+$ – арифметическое сложение



а



б



Схемы наращивания АЛУ при последовательном (а) и параллельном (б) переносах и реализация функций компаратора для группы АЛУ (в)

Несмотря на разнообразие операций, выполняемых АЛУ, его функциональность ограничена: устройство выполняет только операции над двумя переменными, отсутствуют арифметические операции умножения и деления и т. п. Преодоление этой ограниченности достигнуто в *микроспроцессорах*,

Микропроцессор - программно-управляемое устройство для обработки цифровой информации и управления этим процессом.

Характеристики:

□ Разрядность.

□ Производительность.

□ Система команд.

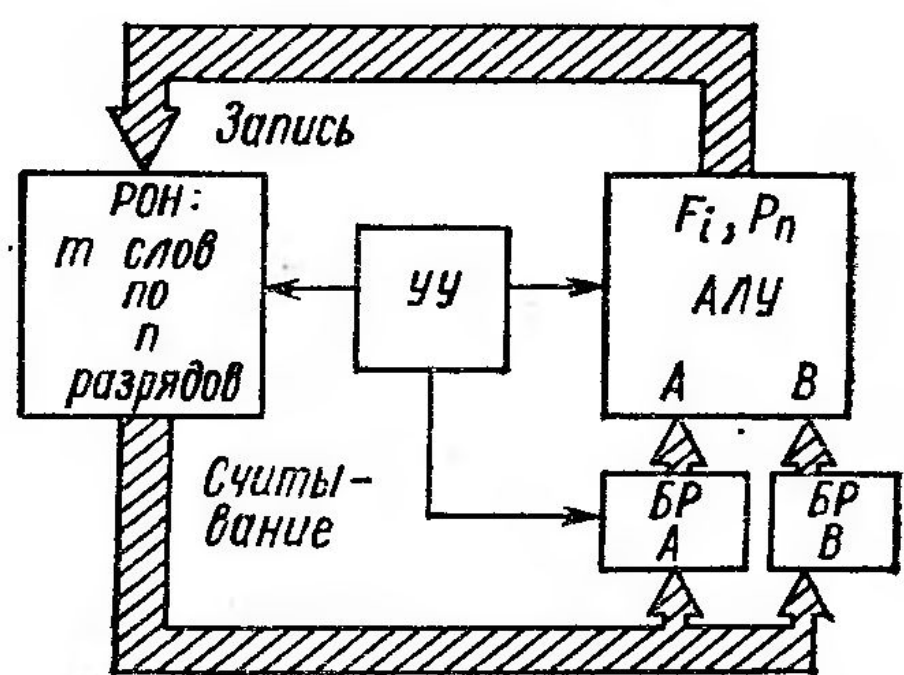
□ Объем адресуемой памяти.

Группы микропроцессоров:

✓ **CISC** - полный набор команд.

✓ **RISC** - сокращённый набор команд.

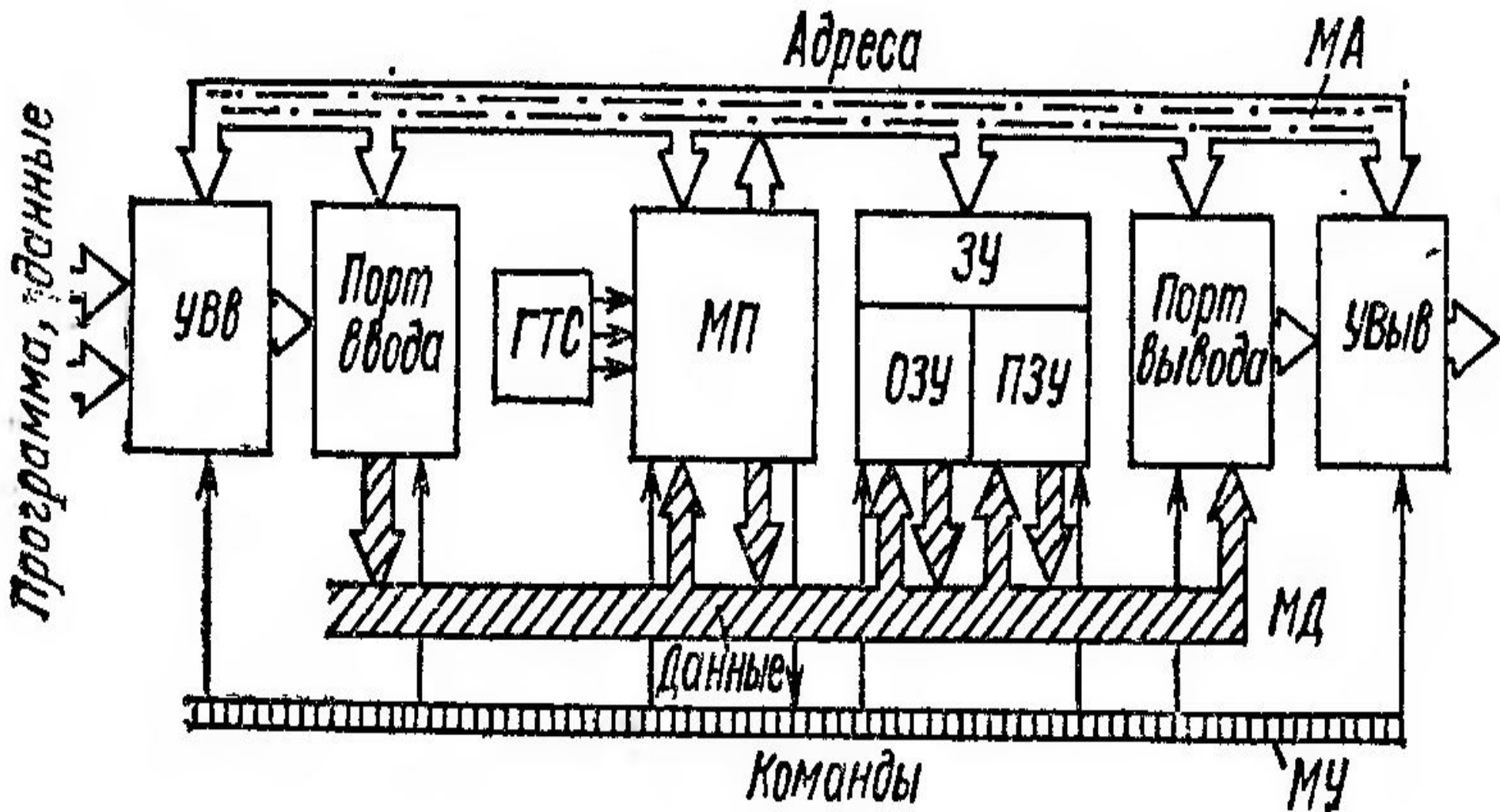
✓ **MISC** - минимальный набор команд, очень высокое быстродействие



МП состоит из двух частей:

1. Операционная: устройства управления, арифметико-логическую и микропроцессорную память.
2. Интерфейсная: адресные регистры, схемы управления портами и шиной, а также блок команд.

Читают и дешифруют команды, поступающие из основной памяти. Принимают и обрабатывают запросы от адаптеров про обслуживание внешних устройств (ВУ). Читают данные из оперативной памяти и регистров ВУ. Создают управляющие сигналы, с помощью которых управляют другими узлами и блоками персонального компьютера. Обрабатывают данные и производят их запись в оперативную память, а также регистры внешних устройств.



Соответствие процессоров корпораций Intel и AMD

Количество ядер	Intel	AMD
1	Pentium 4, Celeron D	Athlon 64
2	Core 2 Duo	Athlon 64 X2
3	—	Phenom X3
4	Core 2 Quad	Phenom X4
6	Core i7 980X	Phenom II X6

Современные процессоры AMD

Сокет	Процессор	Частота, ГГц	Кэш L2, Кбайт	Кэш L3, Мбайт
AM3	Phenom II X2 550	3,2	2×512	6,144
AM3	Phenom II X4 945	3,0	4×512	6,144
AM3	Phenom II X6 1055	2,8	6×512	6,144
AM3	Athlon II X3 450	3,2	3×512	—

Современные процессоры Intel

Сокет	Процессор	Частота, ГГц	Кэш L2, Кбайт	Кэш L3, Мбайт
1156	Pentium G6000	2,8	2×256	3,072
1156	Core i3 550	3,2	2×256	4,096
1156	Core i5 750	3,33	4×1024	8,192
1156	Core i7 960	3,2	4×256	8,192
1156	Xeon X3470	2,93	4×256	8,192



**Матричные
умножители**

- Микросхемы множительных устройств появились в 1980-х годах, когда достигнутый уровень интеграции позволил разместить на одном кристалле достаточно большое количество логических элементов.
- Структура матричных умножителей тесно связана со структурой математических выражений, описывающих операцию умножения.
- Пусть имеются два целых двоичных числа без знаков $A_m = a_{m-1} \dots a_0$ и $V_n = b_{n-1} \dots b_0$, Их перемножение выполняется по известной схеме "умножения столбиком". Если числа четырехразрядные, т. е. $m = n = 4$, то

				×	a_3	a_2	a_1	a_0
					b_3	b_2	b_1	b_0
					a_3b_0	a_2b_0	a_1b_0	a_0b_0
+			a_3b_1	a_2b_1	a_1b_1	a_0b_1		
		a_3b_2	a_2b_2	a_1b_2	a_0b_2			
	a_3b_3	a_2b_3	a_1b_3	a_0b_3				
	p_7	p_6	p_5	p_4	p_3	p_2	p_1	p_0

- Произведение выражается числом $P_{m+n} = P_{m+n-1} P_{m+n-2} \dots P_0$.
- Члены вида $a_i b_j$, где $i = 0 \dots (m-1)$ и $j = 0 \dots (n-1)$ вырабатываются параллельно во времени конъюнкторами. Их сложение в столбцах, которое можно выполнять разными способами, составляет основную операцию для умножителя и определяет почти целиком время перемножения.
- Матричные перемножители могут быть просто множительными блоками (МБ) или множительно-суммирующими (МСБ), последние обеспечивают удобство наращивания размерности умножителя.
- МСБ реализует операцию $P = A_m \times B_n + C_m + D_n$, т. е. добавляет к произведению два слагаемых: одно разрядности m , совпадающей с разрядностью множимого, другое разрядности n , совпадающей с разрядностью множителя.



Множительно- суммирующие блоки

- Множительно-суммирующий блок для четырехразрядных операндов без набора конъюнкторов, вырабатывающих члены вида $a_i b_j$, показан на рис. 2.37, а, где для одноразрядного сумматора принято обозначение (рис. 2.37, б).
- Для построения МСБ чисел равной разрядности потребовалось n^2 конъюнкторов и n^2 одноразрядных сумматоров.

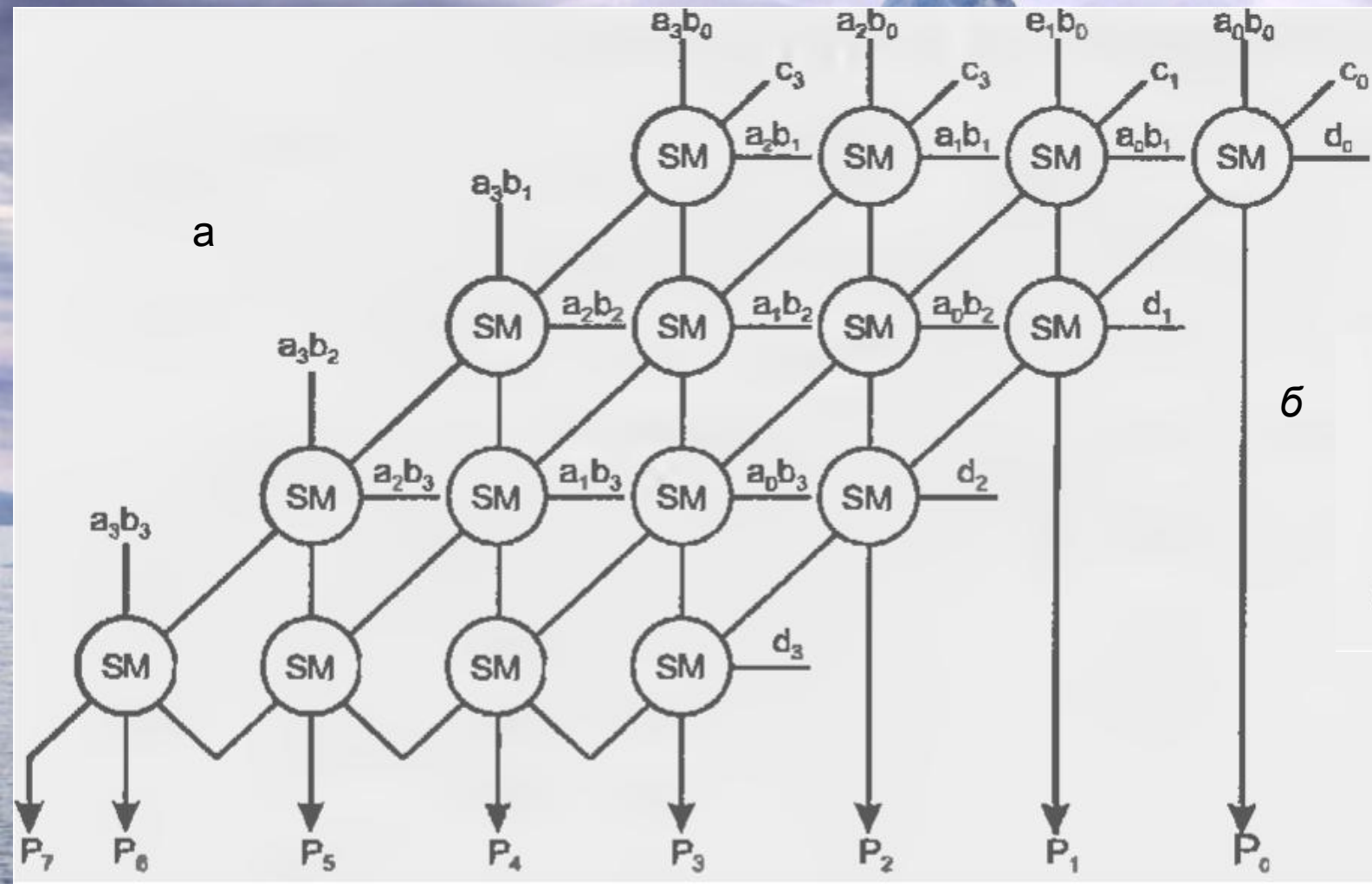


Рисунок 10.3 Схема множително-суммирующего блока для четырехразрядных сомножителей (а), обозначение одноразрядного сумматора для данной схемы (б)

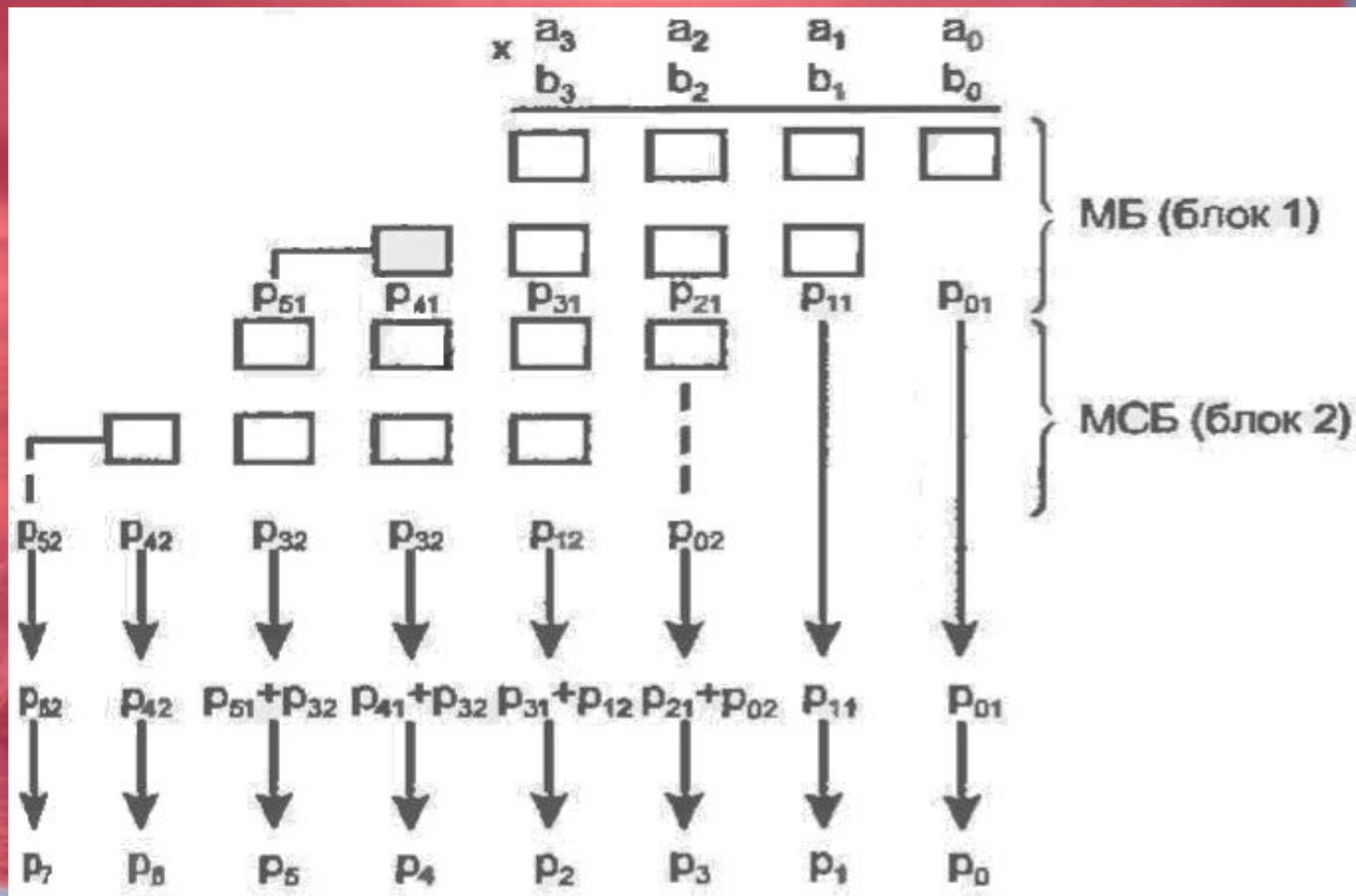


Рисунок 10.4 К пояснению принципа наращивания размерности множительных устройств (v), условное обозначение множително-суммирующего блока

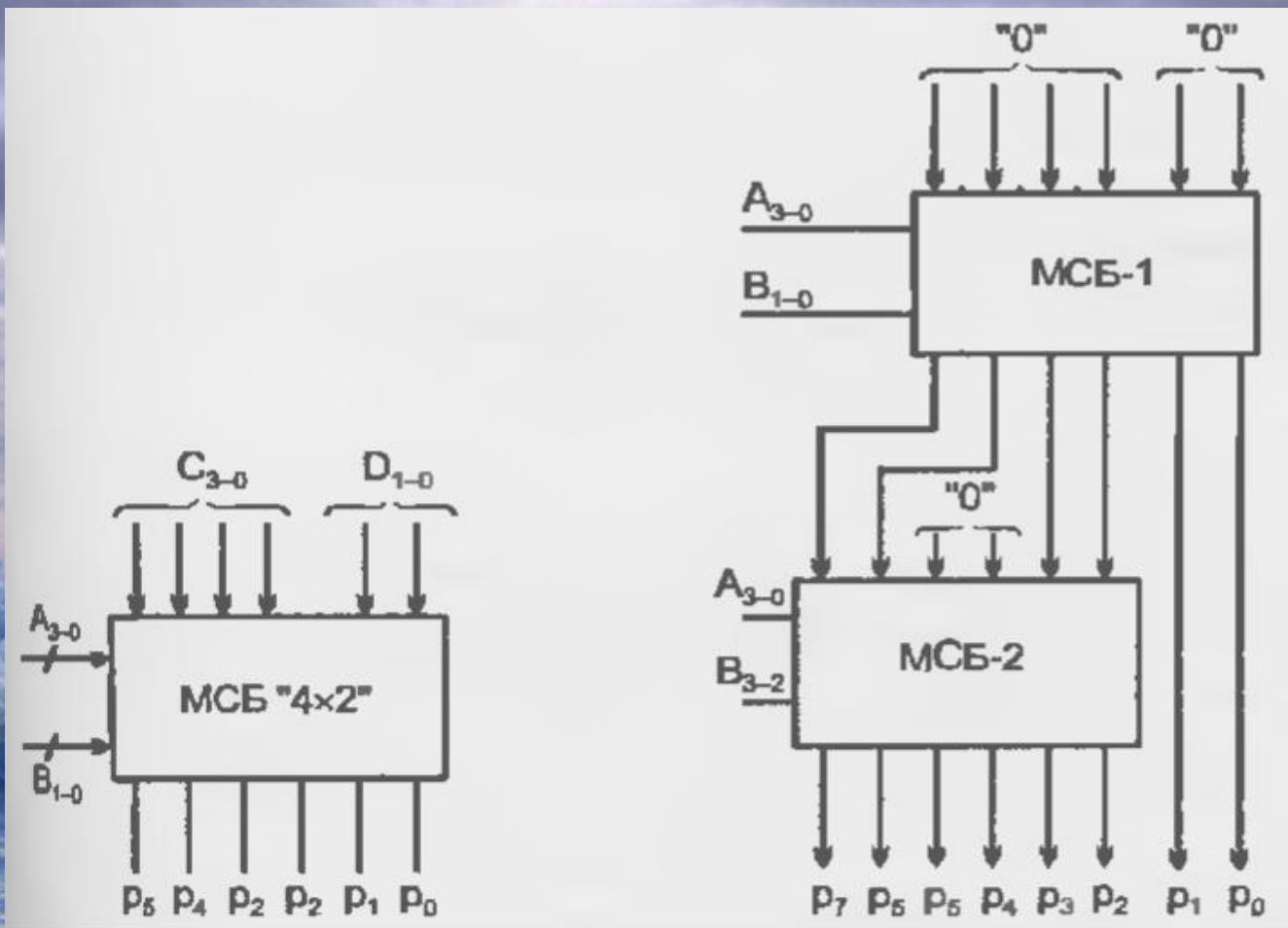


Рисунок 10.5 схема умножителя 4x4",
 построенная на множительно-
 суммирующих блоках "4 x 2" (в)

- Максимальная длительность умножения — сумма задержек сигналов в конъюнкторах для выработки членов a_{ij} и задержки в наиболее длинной цепочке передачи сигнала в матрице одноразрядных сумматоров, равной $2n - 1$ ($m + n - 1$ в общем случае). Таким образом, $t_{MPL} = tk + (2n - 1)t_{sm}$,
- Схема множительного блока отличается от схемы МСБ тем, что в ней отсутствуют сумматоры правой диагонали, т. к. при $C_m = 0$ и $D_n = 0$ они не требуются.
- Построение умножителей большей размерности из умножителей меньшей размерности на основе МБ требует введения дополнительных схем, называемых "деревьями Уоллеса", которые имеются в некоторых зарубежных сериях. При использовании МСБ дополнительные схемы не требуются. Принцип наращивания размерности умножителя иллюстрируется на рис. 2.38, а на примере построения МРЬ "4 x 4" из МСБ "4 x 2". На поле частичных произведений выделены зоны, воспроизведение которых возможно на блоках размерности 4x2 (это две первые строки и две последние).
- Перемножение в пределах зон дает частичные произведения $p_1 = P_{51}P_{41}P_{31}P_{21}P_{11}P_{01}$ и $P_2 = P_{52}P_{42}P_{32}P_{22}P_{12}P_{02}$. Для получения конечного значения произведения эти частичные произведения нужно сложить с учетом их взаимного положения (сдвига одного относительно другого).
- Схема, реализующая указанный принцип, изображена на рис. 2.38, в. В ней использовано условное обозначение МСБ (рис. 2.38, б). Для общности оба блока размерности 4x2 показаны как МСБ, хотя первый может быть просто множительным блоком, т. к. для него слагаемые C и D имеют нулевое значение.

Схемы ускоренного умножения

- Для ускорения умножения разработан ряд алгоритмов, большой вклад в эти разработки внес Э. Бут (E. Boot). Рассмотрим процесс умножения по так называемому модифицированному алгоритму Бута (*умножение сразу на два разряда*).
- Из изложенного выше видно, что основную задержку в процесс выработки произведения вносит суммирование частичных произведений. Уменьшение их числа сократило бы время суммирования. К этому приводит алгоритм, основанный на следующих рассуждениях.
- Пусть требуется вычислить произведение

$$P = A \times B = A \times (b_{n-1} 2^{n-1} + b_{n-2} 2^{n-2} + \dots + b_0 2^0). \quad (a)$$

- Непосредственное воспроизведение соотношения (а) связано с выработкой частичных произведений вида Ab_j2^i ($i = 0 \dots n - 1$). Число таких произведений равно разрядности множителя n .
- Выражение (а) можно видоизменить с помощью соотношения

$$b_i 2^i = b_i 2^{i+1} - 2b_i 2^{i-1}, \quad (б)$$

справедливость которого очевидна.

Это соотношение позволяет разреживать последовательность (спектр) степеней в сумме частичных произведений. Можно, например, исключить четные степени, как показано на рис. 2.39, а. Исключение четных (или нечетных) степеней не только изменяет значения оставшихся частичных произведений, но и сокращает их число примерно вдвое, что, в конечном счете, ускоряет выработку произведения. Для того чтобы "разнести по соседям" член со степенью 2^i , расширим разрядную сетку, введя слагаемое

$b_{i-1} 2^{i-1}$ (нулевой разряд с номером -1).

Оставшиеся частичные произведения имеют вид

$$R_i = A(-2b_{i+1} + b_i + b_{i-1})2^i.$$

Так как число частичных произведений уменьшилось примерно вдвое, при применении этого алгоритма говорят об умножении сразу на два разряда.

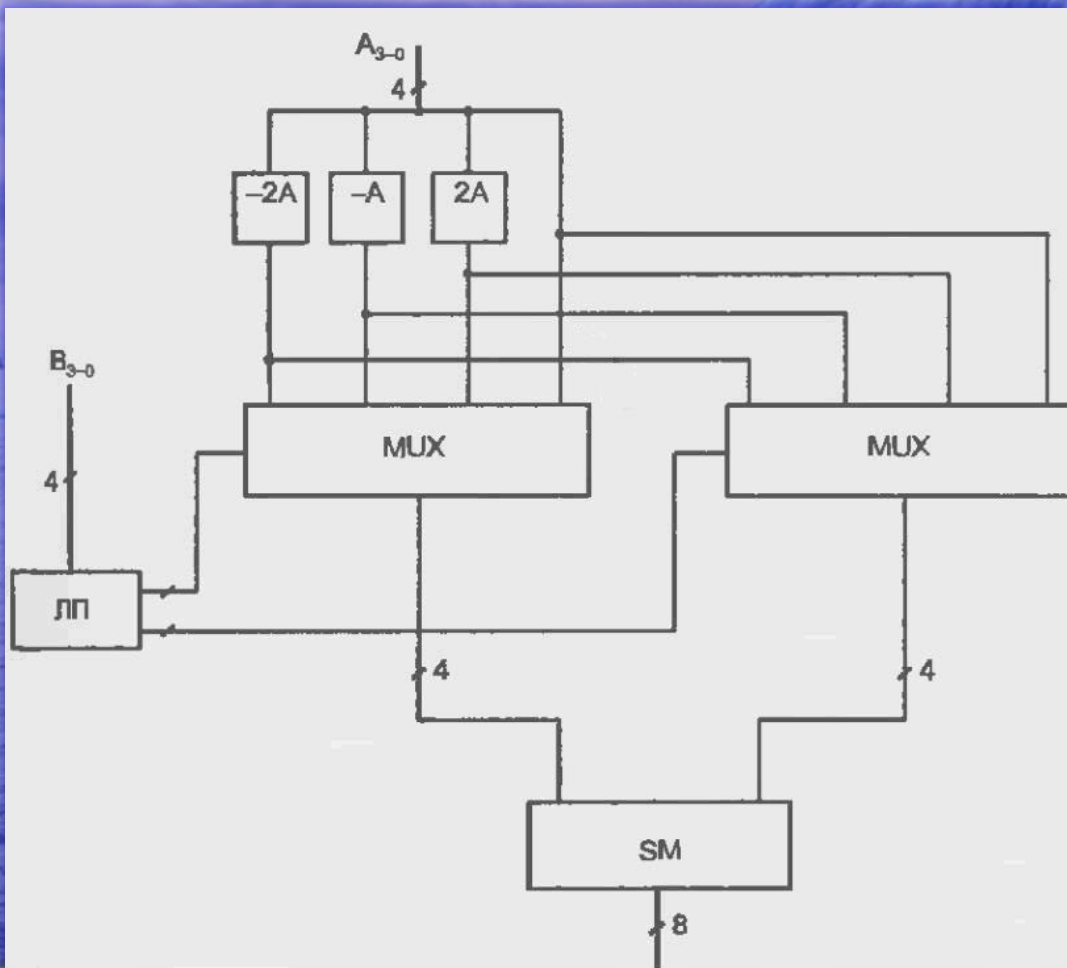
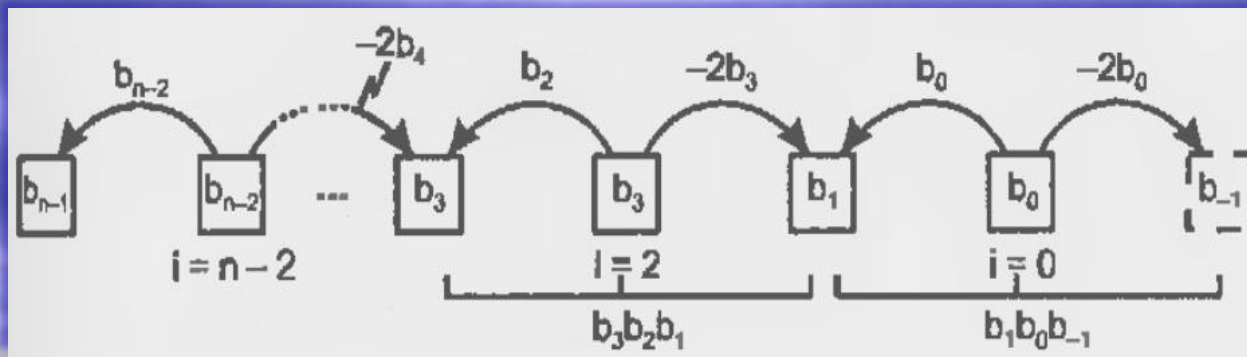


Рисунок 10.6 К пояснению принципа быстрого умножения "среза на два разряда" (в) и схема быстрого умножения (б)

Для всех возможных сочетаний b_{i+1} , b_i , b_{i-1} можно составить таблицу (табл. 2.14) частичных произведений.

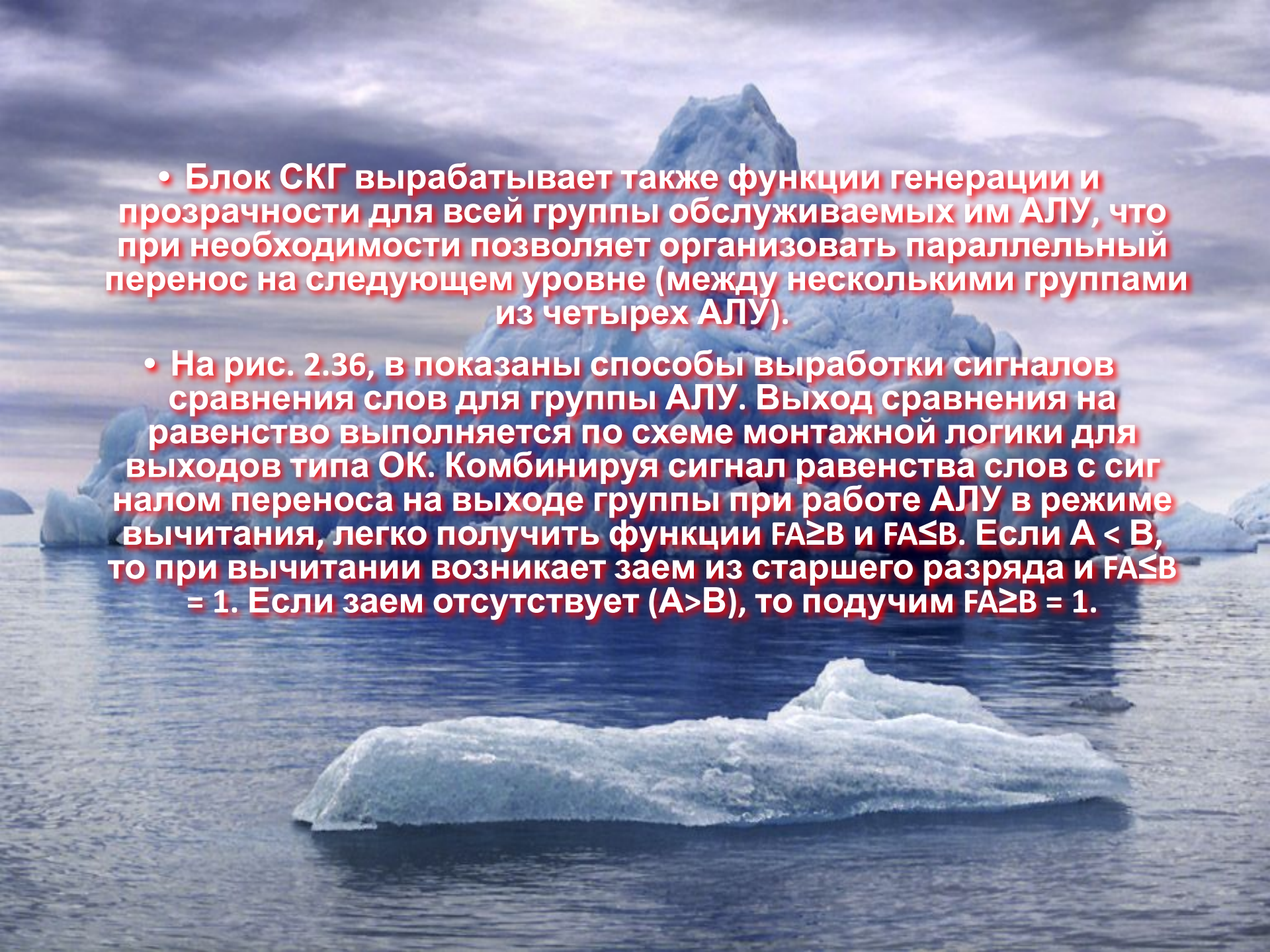
b_{i+1}	b_i	b_{i-1}	Значение скобки	$R_i/2^i$	Операция для получения $R_i/2^i$
0	0	0	0	0	Заменить A нулем
0	0	1	1	A	Скопировать A
0	1	0	1	A	Скопировать A
0	1	1	2	2A	Сдвинуть A влево
1	0	0	-2	-2A	Сдвинуть A влево и преобразовать в дополнительный код
1	0	1	-1	-A	Преобразовать A в дополнительный код
1	1	0	-1	-A	Преобразовать A в дополнительный код
1	1	1	0	0	Заменить A нулем

Пример

- Пусть требуется умножить 10102 на 01112, т. е. 10 x 7. При разреживании частичных произведений оставим только нечетные, как показано на рис. 2.39, а. Расширив разрядную сетку множителя, имеем $B = b_4b_3b_2b_1b_0b_{-1}b_{-2} = 0011100$.
- Первому частичному произведению соответствует тройка $b_0b_{-1}b_{-2} = 100$. Из табл. 2.14 получаем, что этой тройке соответствует частичное произведение $-2A_{2-1} = -A$, для получения которого требуется перевести A в дополнительный код. Сама величина A в пределах разрядной сетки произведения должна быть записана как 00001010, ее обратный код 11110101 и дополнительный код 11110110.
- Второму частичному произведению соответствует тройка $b_2b_1b_0 = 111$, следовательно, второе частичное произведение равно нулю (табл. 2.14).
- Третьему частичному произведению соответствует тройка $b_4b_3b_2 = 001$, следовательно, оно имеет вид $A_{23} = 01010000$.
- Для получения результата заданного умножения требуется сложить частичные произведения:

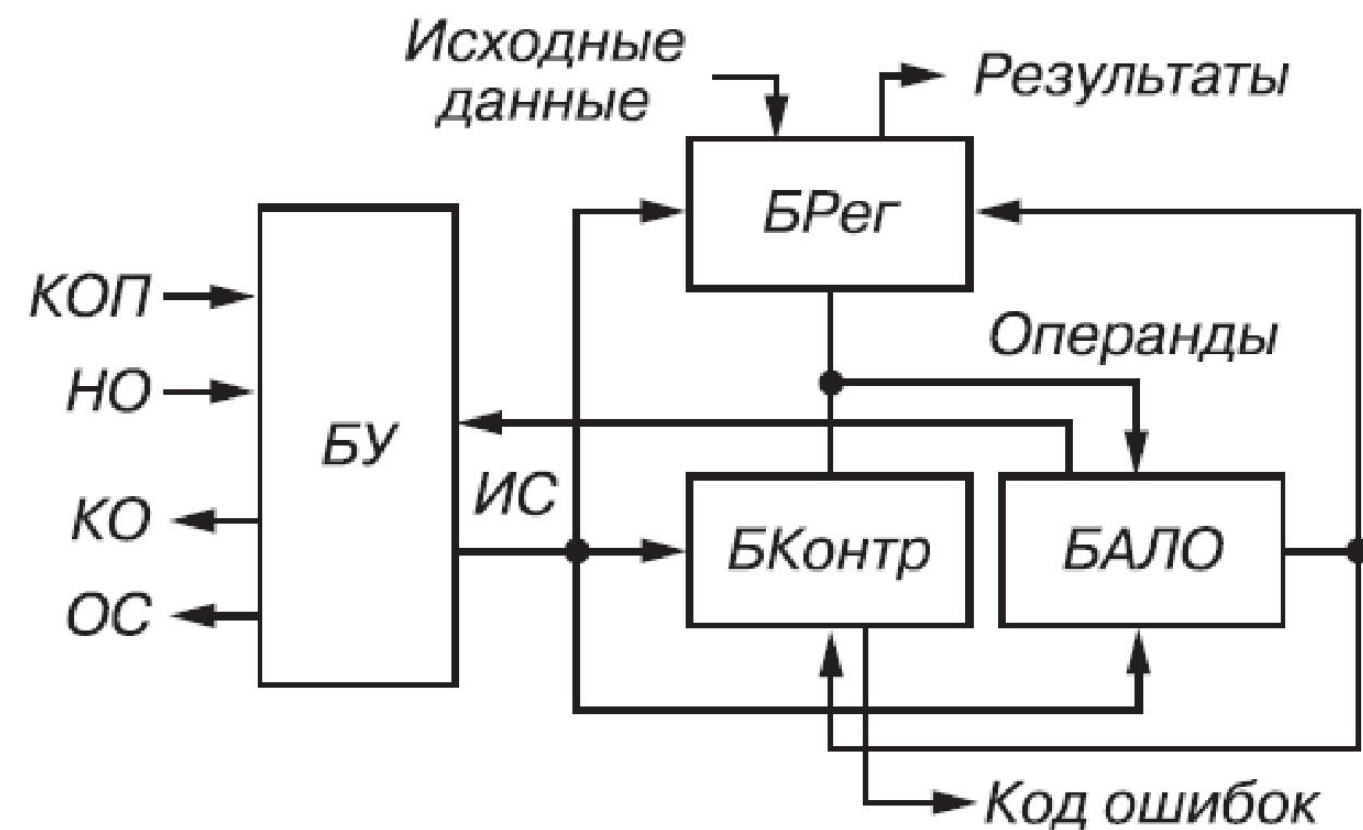
$$\begin{array}{r} 11110110 \\ 01010000 \\ \hline 01000110 = 2^6 + 2^2 + 2^1 = 64 + 4 + 2 = 70. \end{array}$$

- Схема, реализующая алгоритм быстрого умножения сразу на два разряда, показана на рис. 2.39, б.
- Множимое А поступает в этой схеме на ряд преобразователей, заготавливающих все возможные варианты частичных произведений ($-2A$, $-A$, $2A$), кроме самого А и нуля, которые не требуют схемной реализации. Множитель В поступает на логический преобразователь ЛП, который анализирует тройки разрядов, декодирует их и дает мультиплексорам сигналы выбора того или иного варианта частичных произведений. Окончательный результат получается суммированием частичных произведений с учетом их взаимного сдвига в разрядной сетке. Размерность умножителя "4 x 4".
- Приведенные выше примеры множительных устройств касались операций с прямыми кодами. В этом случае умножение знакопеременных чисел сведется только к выработке знакового разряда как суммы по модулю 2 знаковых разрядов сомножителей. Если же числа представлены не прямыми кодами с знаковыми разрядами, а, например, дополнительными кодами, то, имея рассмотренные выше умножители, можно дополнить их преобразователями дополнительного кода в прямые на входах и преобразователем прямого кода в дополнительный на выходе или использовать схемы, непосредственно реализующие алгоритмы умножения дополнительных кодов (см., например, [37]).
- Разработке матричных умножителей уделяют внимание многие фирмы. В отечественных сериях МИС/СИС имеются умножители малой размерности (2 x 2, 4 x 4, 4 x 2 и др.). В сериях БИС размерности умножителей значительно больше. В серии 1802, например, имеются умножители 8x8, 12x12, 16x16 (BP3, BP4 и BP5 соответственно). В схемотехнике ЭСЛ выполнен умножитель 1800BP1 (8 x 8 за 17 нс). Зарубежные фирмы разработали умножители (фирмы ВТ, Hitachi и др.) размерностями 16 x 16 и более с временами умножения 3...5 нс. Несколько лет назад предприятие "Интеграл" (г. Минск) выпустило умножитель КА1843BP1 размерностью 32 x 32 со временем умножения 250 нс в корпусе с 172 выводами.

A photograph of several icebergs floating in a dark blue sea under a cloudy sky. The icebergs are white and blue, with some showing sharp peaks and others being more rounded. The water is calm, and the sky is overcast.

- **Блок СКГ вырабатывает также функции генерации и прозрачности для всей группы обслуживаемых им АЛУ, что при необходимости позволяет организовать параллельный перенос на следующем уровне (между несколькими группами из четырех АЛУ).**

- **На рис. 2.36, в показаны способы выработки сигналов сравнения слов для группы АЛУ. Выход сравнения на равенство выполняется по схеме монтажной логики для выходов типа ОК. Комбинируя сигнал равенства слов с сигналом переноса на выходе группы при работе АЛУ в режиме вычитания, легко получить функции $FA \geq B$ и $FA \leq B$. Если $A < B$, то при вычитании возникает заем из старшего разряда и $FA \leq B = 1$. Если заем отсутствует ($A > B$), то подучим $FA \geq B = 1$.**



- ✓ блок регистров *БРег*, предназначенный для приёма и размещения операндов и результата операции;
- ✓ блок арифметико-логических операций *БАЛО*, в котором осуществляется преобразование операндов согласно коду операции (*КОП*) в реализуемой команде;
- ✓ блок контроля *БКонтр*, обеспечивающий непрерывный оперативный контроль и диагностику ошибок;
- ✓ блок управления *БУ*, в котором формируются импульсы синхронизации *ИС*, координирующие взаимодействие всех блоков АЛУ между собой и с другими блоками процессора.

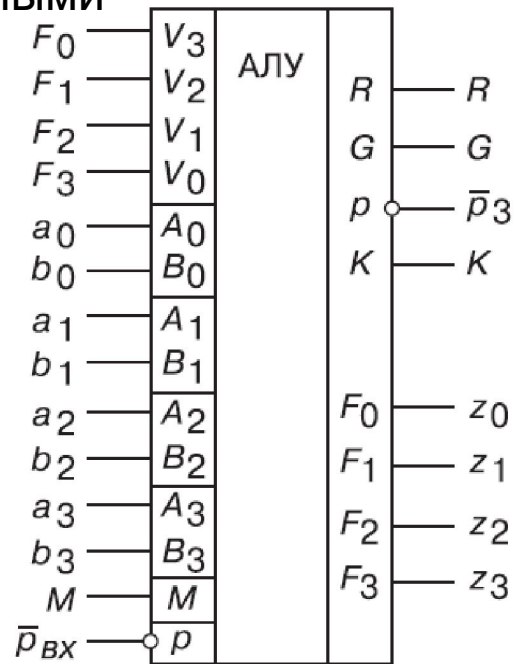
Устройство работает в соответствии с сообщаемыми ему кодами операций, которые нужно выполнить над переменными, помещаемыми в регистры. На разных этапах выполнения команды операции производится анализ преобразований информации, и на основании сигналов признаков ПР блок БУ формирует и выдаёт осведомительный сигнал ОС, характеризующий некоторое состояние процессора. В асинхронных АЛУ выполнение операции производится по сигналу НО (начало операции), а переход к выполнению очередной команды — по сигналу КО (конец операции).

В основе построения схем АЛУ положены одноразрядные комбинационные сумматоры, образующие при определённой коммутации 4разрядный сумматор с параллельным переносом.

Каждая схема АЛУ обеспечивает выполнение над 4-разрядными словами A и B шестнадцати поразрядных логических и арифметико-логических операций (таблица 6.7).

Коммутация АЛУ на выполнение той или иной из 16 операций в заданном режиме производится управляющими сигналами $V_3V_2V_1V_0$ и сигналом M .

При $M = 0$ АЛУ выполняет арифметические операции, а при $M = 1$ — логические. Комбинация входных сигналов $V_3V_2V_1V_0$ определяет одну из 16 выполняемых функций.



При подаче на информационные входы $A_3A_2A_1A_0$ и $B_3B_2B_1B_0$ 4-разрядных двоичных чисел $a_3a_2a_1a_0$ и $b_3b_2b_1b_0$ результат суммирования $z_3z_2z_1z_0$ фиксируется на выходах $F_3F_2F_1F_0$, а результат сравнения — на выходе K .

В каждом из четырёх разрядов АЛУ значения z_i ($i = 0, 1, 2, 3$) формируются схемами, реализующими переключательную функцию

$$z_i = s_i^* \bar{p}_i \vee \bar{s}_i^* p_i,$$

где $s_i^* = \bar{T}_i^* \bar{G}_i^* \vee \bar{T}_i^* G_i^*$; $\bar{T}_i^* = \overline{a_i \vee bV_0 \vee \bar{b}_i V_1}$; $\bar{G}_i^* = a_i \bar{b}_i V_2 \vee a_i b_i V_3$.

Сигналы переноса внутри АЛУ из младшего разряда в старший формируются схемой параллельного переноса под управлением сигнала M :

$$p_{ex}^i = p_{ex} \vee M; \quad p_0 = G_0 \vee T_0 p_{ex} \vee M; \quad p_1 = G_1 \vee T_1 G_0 \vee T_1 T_0 p_{ex} \vee M;$$

$$p_2 = G_2 \vee T_2 G_1 \vee T_2 T_1 G_0 \vee T_2 T_1 T_0 p_{ex} \vee M,$$

а сигнал переноса из старшего разряда p_3 формируется схемой, реализующей выражение

$$\bar{p}_3 = \bar{G} \vee \bar{p}_{ex} R_i,$$

где $\bar{G} = \overline{\bar{T}_3 \vee \bar{T}_2 \bar{G}_3 \vee \bar{T}_1 \bar{G}_2 \bar{G}_3 \vee \bar{T}_0 \bar{G}_1 \bar{G}_2 \bar{G}_3}$ — функция генерации переноса из АЛУ;
 $R = \bar{G}_0 \bar{G}_1 \bar{G}_2 \bar{G}_3$ — функция распространения переноса через 4-разрядную группу сумматора АЛУ.

Пользуясь приведенными выражениями, можно проанализировать процессы выполнения всех операций, перечисленных в таблице 6.7. Например, при $M = 1$ и $V_3V_2V_1V_0 = 0000$ имеем $\overline{T}_i^* = a_i$, $\overline{G}_i^* = 0$, $\overline{s}_i^* = a_i$ и $z_i = a_i$, т. е. схема АЛУ выполняет операцию отрицания числа $A = a_3a_2a_1a_0$; при $M = 0$ и $V_3V_2V_1V_0 = 1001$ — арифметическое сложение $A + B$, при $V_3V_2V_1V_0 = 0110$ — вычитание $A - B$, а при $V_3V_2V_1V_0 = 1100$ — сдвиг влево на один разряд и $\overline{p}_{ex} = 0$.

Результат сравнения двух 4-разрядных чисел A и B на выходе K (см. рис. 6.28) формируется схемой АЛУ в соответствии с выражением

$$K = \overline{z_0 \vee z_1 \vee z_2 \vee z_3}.$$

Для организации параллельного переноса данных между каскадами АЛУ предусмотрены выходы: G — выход генерации переноса; R — выход распространения переноса и p — выход переноса со старшего разряда АЛУ.

- Арифметико-логические устройства АЛУ (ALU, Arithmetic-Logic Unit) выполняют над словами ряд действий. Основой АЛУ служит сумматор, схема которого дополнена логикой, расширяющей функциональные возможности АЛУ и обеспечивающей его перестройку с одной операции на другую
- Обычно АЛУ четырехразрядны и для наращивания разрядности объединяются с формированием последовательных или параллельных переносов. Логические возможности АЛУ разных технологий (ТТЛШ, КМОП, ЭСЛ) сходны. В силу самодвойственности выполняемых операций условное обозначение и таблица истинности АЛУ встречаются в двух вариантах, отличающихся взаимно инверсными значениями переменных
- АЛУ (рис. 2.35) имеет входы операндов А и В, входы выбора операций S, вход переноса C_i и вход M (Mode), сигнал которого задает тип выполняемых операций: логические ($M = 1$) или арифметико-логические ($M = 0$). Результат операции вырабатывается на выходах F, выходы G и H дают функции генерации и прозрачности, используемые для организаций параллельных наращивания размерности АЛУ. Сигнал C_o — выходной перенос

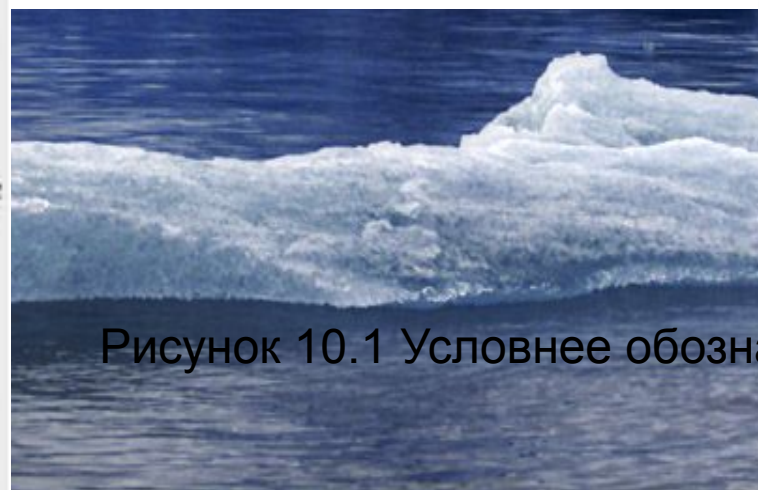
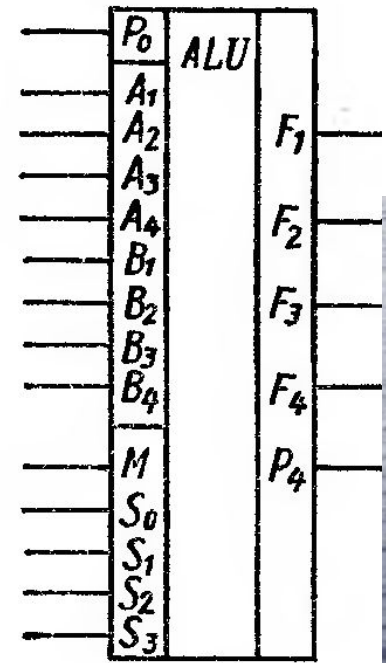
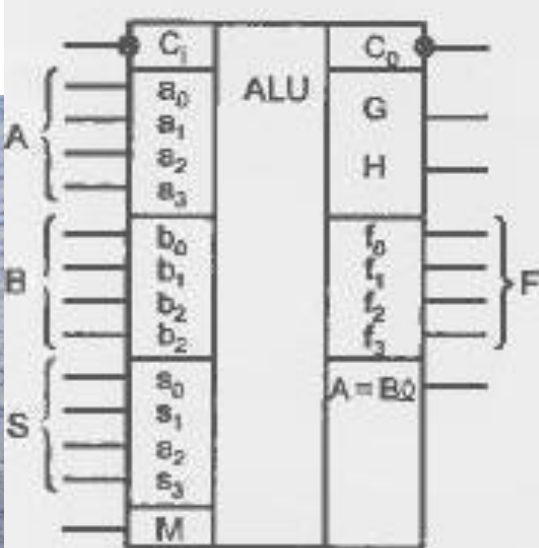


Рисунок 10.1 Условное обозначение АЛУ

- Шестнадцать логических операций позволяют воспроизводить все функции двух переменных. В логико-арифметических операциях встречаются и логические и арифметические операции одновременно.
- Запись типа $A \setminus B + AB$ следует понимать так: вначале поразрядно выполняются операции инвертирования (B), логического сложения ($A \setminus B$) и умножения (AB), а затем полученные указанным образом два четырехразрядных числа складываются арифметически.
- При операциях над словами большой размерности АЛУ соединяются друг с другом с организацией последовательных (рис. 2.36, а) или параллельных (рис. 2.36, б) переносов. В последнем случае совместно с АЛУ применяют микросхемы — блоки ускоренного переноса (CRU, Carry Unit), получающие от отдельных АЛУ функции генерации и прозрачности, а также входной перенос и вырабатывающие сигналы переноса

Таблица 4.16. Логические операции, выполняемые арифметико-логическим устройством (АЛУ)

№	Сигналы управления				Логическая операция
	S_3	S_2	S_1	S_0	
1	0	0	0	0	$F = \bar{A}$
2	0	0	0	1	$F = \overline{A+B}$
3	0	0	1	0	$F = \bar{A}B$ <u>запрет по A</u>
4	0	0	1	1	$F = 0$
5	0	1	0	0	$F = \bar{A}+\bar{B}$
6	0	1	0	1	$F = \bar{B}$
7	0	1	1	0	$F = \bar{A}\bar{B}+\bar{A}B$
8	0	1	1	1	$F = \bar{A}\bar{B}$
9	1	0	0	0	$F = \bar{A}+B$ <u>импликация по A</u>
10	1	0	0	1	$F = AB+\bar{A}\bar{B}$
11	1	0	1	0	$F = B$
12	1	0	1	1	$F = AB$
13	1	1	0	0	$F = 1$
14	1	1	0	1	$F = A+\bar{B}$
15	1	1	1	0	$F = A+B$
16	1	1	1	1	$F = A$

Перечень выполняемых АЛУ операций дан в табл. 2.13. Для краткости двоичные числа $s_3s_2s_1s_0$ представлены их десятичными эквивалентами. Под утолщенными обозначениями 1 и 0 следует понимать наборы 1111 и 0000, входной перенос поступает в младший разряд слова, т. е. равен $000C_i$. Логические операции поразрядные, т. е. операция над словами $A * B$ означает, что $a_i * b_i$; при отсутствии взаимовлияния разрядов. При арифметических операциях учитываются межразрядные переносы.

S	Логические функции (M = 1)	Арифметико-логические функции (M = 0)
0	\bar{A}	$A + C_i$
1	$A \vee B$	$A \vee B + C_i$
2	$\bar{A}B$	$A \vee \bar{B} + C_i$
3	0	$1 + C_i$
4	$\bar{A}B$	$A + AB + C_i$
5	B	$A \vee B + \bar{A}B + C_i$
6	$A \oplus B$	$A + B + C_i$
7	AB	$\bar{A}\bar{B} + 1 + C_i$
8	$A \vee B$	$A + AB + C_i$
9	$\overline{A \oplus B}$	$A + B + C_i$
10	B	$A \vee B + AB + C_i$
11	AB	$AB + 1 + C_i$
12	1	$A + A + C_i$
13	$A \vee B$	$A \vee B + A + C_i$
14	$A \vee B$	$A \vee \bar{B} + A + C_i$
15	A	$A + 1 + C_i$