

Ускоренное умножение

$$t_{\text{умч}} = \sum_{i=1}^{n-1} (t_{\text{сдс}} + p_i t_{\text{сл}}),$$

Методы ускорения умножения можно условно разделить на —

- аппаратные
- логические



АЛГОРИТМ МАК-СОРЛИ С МЛАДШИХ РАЗЯДОВ МНОЖИТЕЛЯ С ОБРАБОТКОЙ ДВУХ РАЗЯДОВ МНОЖИТЕЛЯ ЗА ТАКТ

Триггер	Y_i, Y_{i+1}	Знак арифметич. действия	Кратные множимого
0	00	Нет действий	
0	01	+	1A
0	10	+	2A
0	11	-	1A*
1	00	+	1A
1	01	+	2A
1	10	-	1A*
1	11	Нет действий	



$$9 * (-9) = -81$$

00.00000 00000 11.0111

11.0111 $\tau=0, -1A$

11.0111

11.110111 модиф. сдвиг

01.0010 $\tau=1, +2A$

00.1111110000

00.0011111100 обычный сдвиг

11.0111 $\tau=0 -1A$

11.1010111100

11 1110101111- модиф. сдвиг

Триггер	Y_i, Y_{i+1}	Знак арифметич. действия	Кратные множимого
0	00	Нет действий	
0	01	+	1A
0	10	+	2A
0	11	-	1A*
1	00	+	1A
1	01	+	2A
1	10	-	1A*
1	11	Нет действий	





АЛГОРИТМ МАК-СОРЛИ СО СТАРШИХ РАЗЯДОВ МНОЖИТЕЛЯ С ОБРАБОТКОЙ ДВУХ РАЗЯДОВ МНОЖИТЕЛЯ ЗА ТАКТ

Y_i	$Y_{i+1} Y_{i+2}$	Арифмет. действия	Кратные множимого
0	00	Нет действий	
0	01	+	2А
0	10	+	2А
0	11	+	4А
1	00	-	4А
1	01	-	2А
1	10	-	2А
1	11	Нет действий	



Пример $9 \cdot 9 = 81$

00.00000 00000 **0.1001**

00 00000 10010 +2A

0000000 10010

00.00010 01000сдвиг

0 0000 00 10010+2A

00.00010 11010

11.111 11 10111коррекция -1A

00.00010 10001

Y_i	$Y_{i+1} Y_{i+2}$	Арифмет. действия	Кратные множимого
0	00	Нет действий	
0	01	+	2A
0	10	+	2A
0	11	+	4A
1	00	-	4A
1	01	-	2A
1	10	-	2A
1	11	Нет действий	

0.1001



Алгоритм Лемана

$$\sum_{i=k}^n 2^i = 2^n - 2^k$$

$$0111 = 100\bar{1}$$

На каждом шаге умножения выполняются действия, которые можно описать с помощью логических выражений :

$$\alpha_i = (b_i \oplus b_{i-1}) \bar{\alpha}_{i-1}$$

$$S_i = \alpha_i b_{i+1} \vee S_{i-1} \bar{\alpha}_i$$

α_i - двоичная переменная, единичное значение которой указывает на необходимость выполнения арифметических операций,

b_i - разряды множителя,

S_i - определяет знак арифметической операции,

$S_i = 0$, операция сложения,

$S_i = 1$, операция вычитания.

Основные этапы алгоритма

1. Прием сомножителей, запоминание знаков, формирование модулей.
2. Анализ сомножителей на 0. Формирование знака результата. Установка значения СчЦ.
3. Анализ младшего разряда множителя. Если младший разряд множителя равен 0, то производится сдвиг до появления первой 1. Если младший разряд множителя равен 1, то выполняется вычитание, если следующий разряд 1 и сложение, если следующий разряд 0.
4. Сдвиг суммы ЧП вправо на один разряд прямой или модифицированный сдвиг, если промежуточная сумма частичных произведений отрицательна.. Пункты 3, 4 повторяются для всех цифровых разрядов множителя.
5. Формирование результата.

Пример: Разряды множителя 0111($b_3 b_2 b_1 b_0$)

$$\alpha_0 = (b_0 \oplus b_{-1}) \bar{\alpha}_{-1} = (1 \oplus 0) 1 = 1$$

$$S_0 = \alpha_i b_{i+1} \vee S_{i-1} \bar{\alpha}_i = \alpha_0 b_1 \vee S_{-1} \bar{\alpha}_0 = 1 \cdot 1 \vee 0 \cdot 0 = 1$$



$$10^*7 \quad 01010 * 0.0111$$

Перевод в с.с. $10\bar{1}$ $0.0111 \gg \gg 0.100\bar{1}$

$$\begin{array}{r}
 00000 \ 00000 \quad 0.100\bar{1} \\
 +10110 \quad \leftarrow \\
 \hline
 10110 \ 00000 \\
 11011 \ 00000 \\
 11110 \ 11000 \quad \leftarrow \\
 +01010 \quad \downarrow \\
 \hline
 01000 \ 11000 \\
 00100 \ 01100 \\
 00010 \ 00110 = 2+4+64
 \end{array}$$

ПРИМЕР ПЕРЕВОДА В СИСТЕМУ $10\bar{1}$

$$101110 = 46$$

$$1100\bar{1}0 = 46$$

$$10\bar{1}00\bar{1}0 = 46$$



МАТРИЧНЫЕ МЕТОДЫ УМНОЖЕНИЯ

Каждый элемент $a_i b_j$ ($i, j = 1, n$) принимает значение 0 или 1. Произведение $A \cdot B$ может быть получено, если суммировать элементы матрицы .

$$P = A \cdot B = \left(\sum_{i=0}^{n-1} a_i \cdot 2^i \right) \cdot \left(\sum_{j=0}^{n-1} b_j \cdot 2^j \right) = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} a_i b_j \cdot 2^{i+j}$$



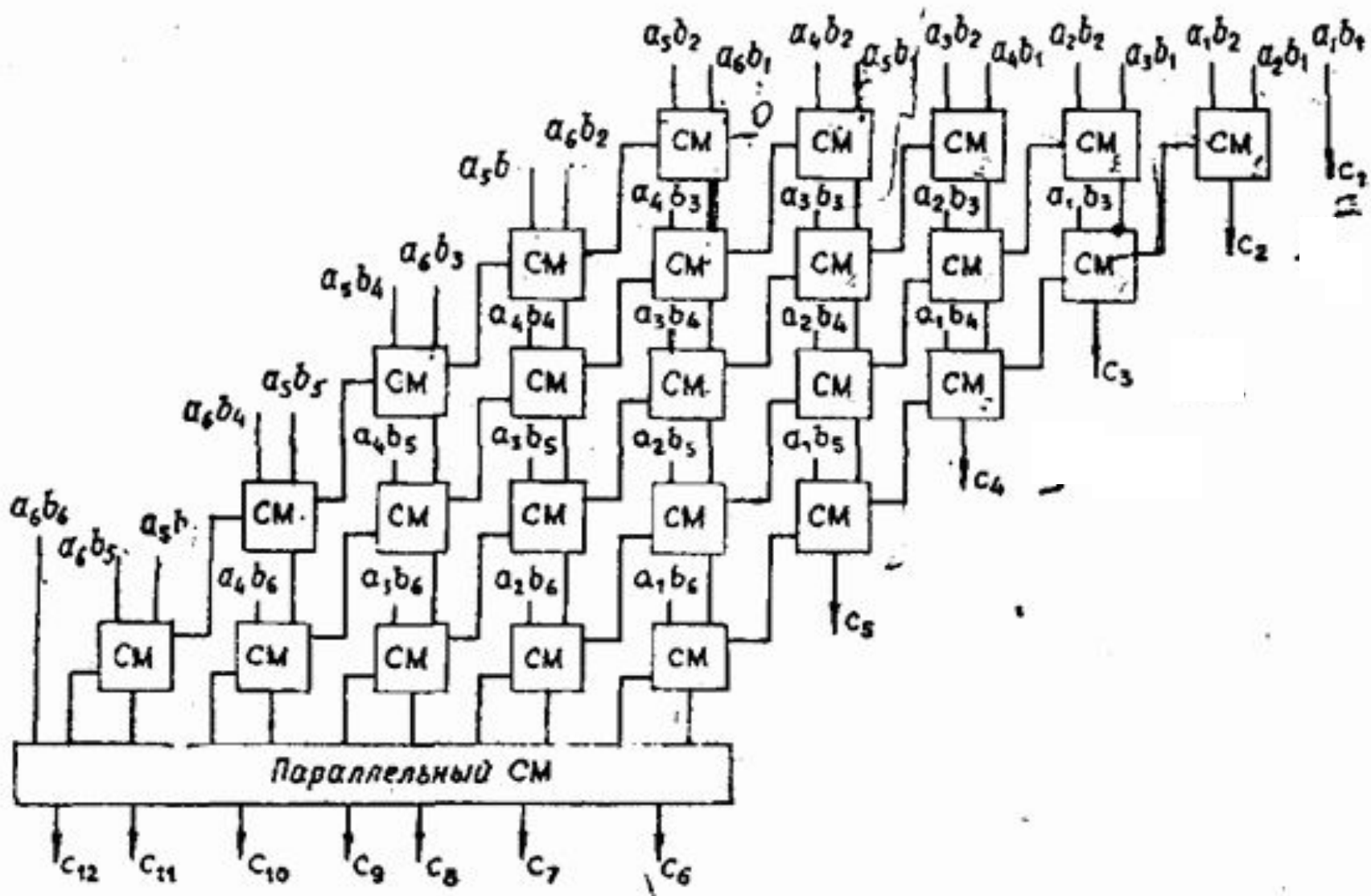


Схема устройства для реализации дерева спуска, реализующая матричный алгоритм умножения

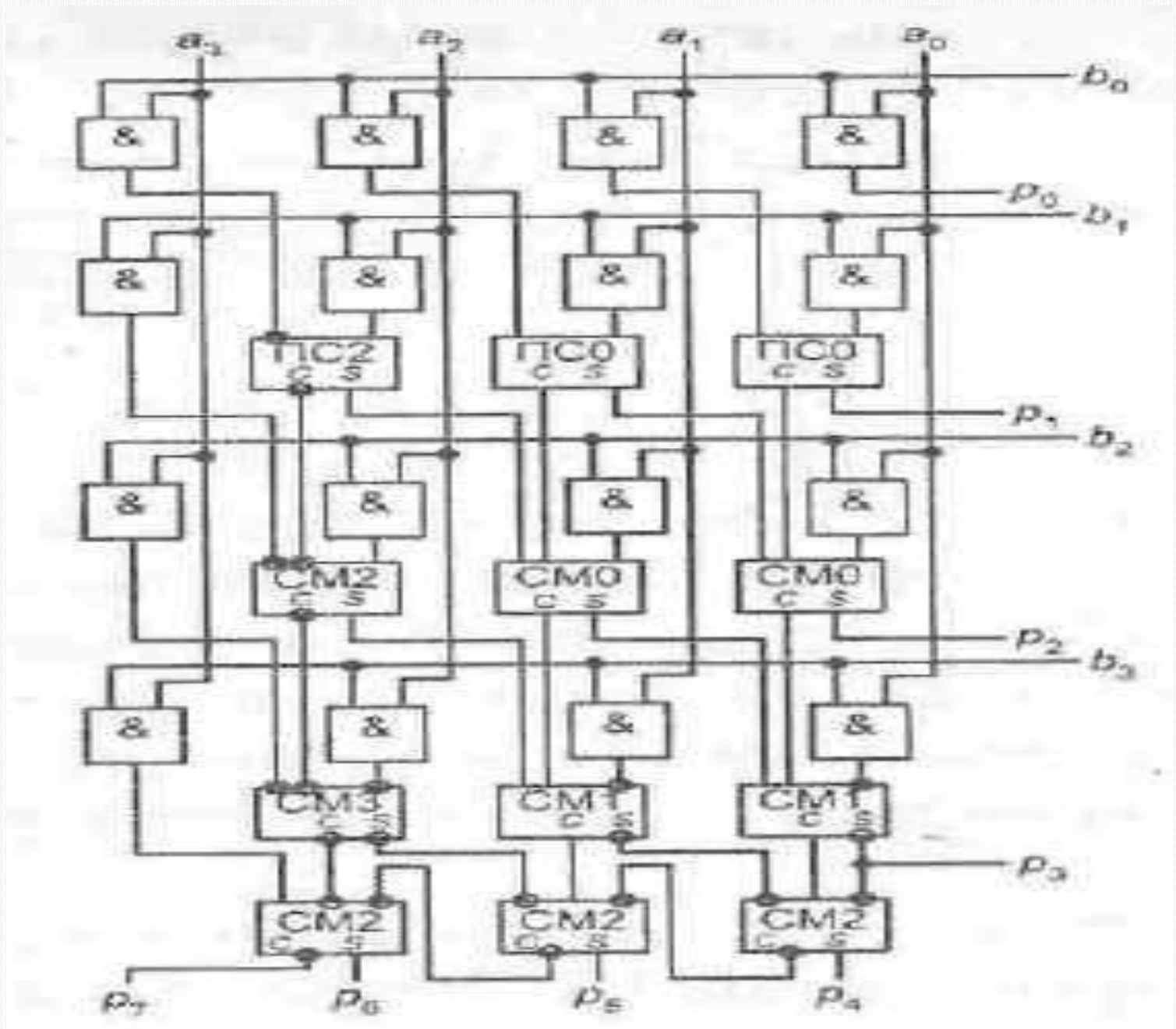


Матричный умножитель $n \times n$ содержит n^2 схем «И», n ПС и $(n^2 - 2n)$ СМ.

Если принять, что для реализации полусумматора требуются два логических элемента, а для полного сумматора — пять, то общее количество логических элементов в умножителе составляет $n^2 + 2n + 5(n^2 - 2n) = 6n^2 - 8n$.

Полагая задержки в схеме «И» и полусумматоре равными θ , а в полном сумматоре — 2θ . общую задержку в умножителе можно оценить выражением $\{4n - 5\}\theta$.

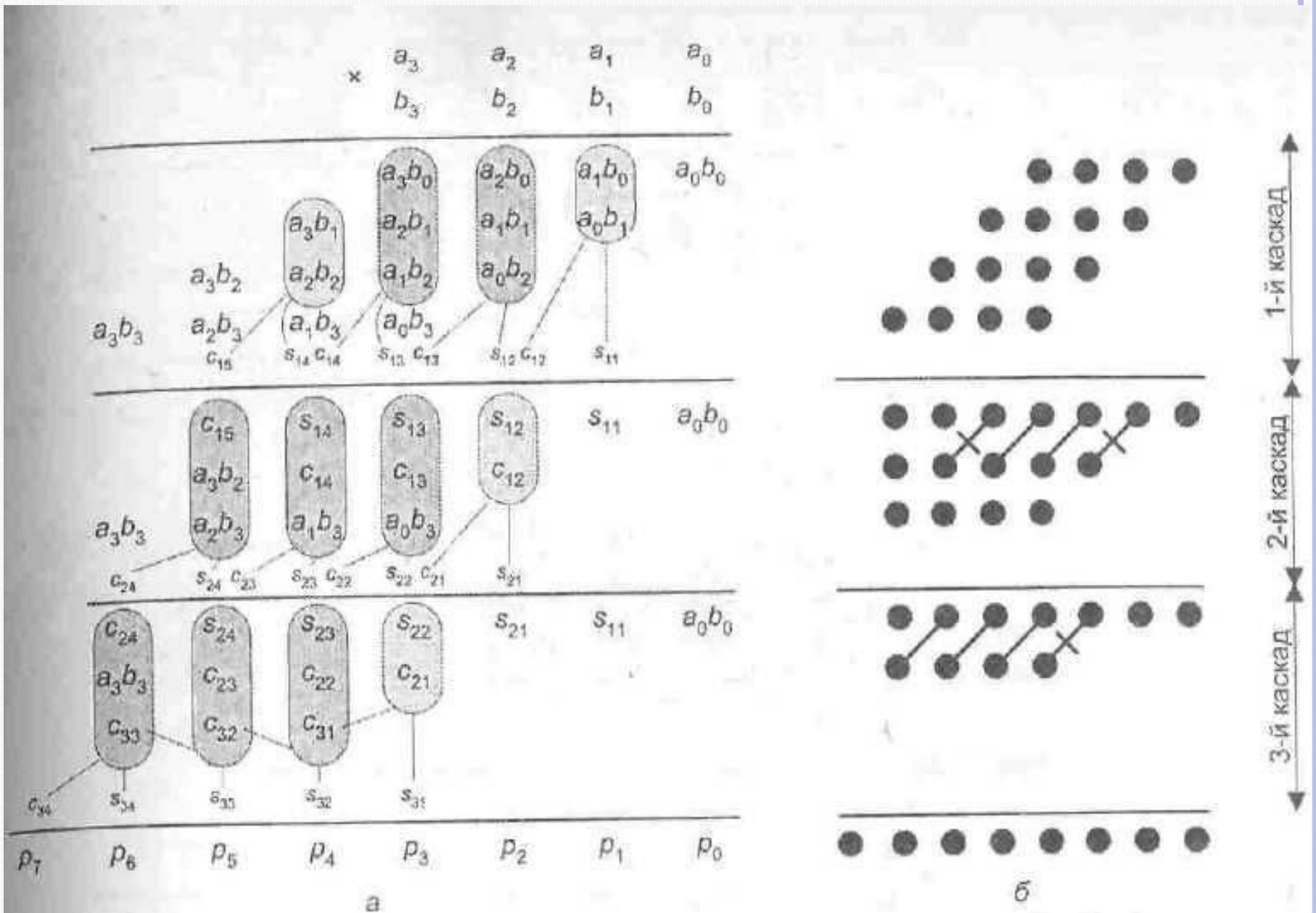




Древовидные умножители включают в себя три ступени:

- ступень формирования битов частичных произведений, состоящую из n^2 элементов «И»;
- ступень сжатия частичных произведений — реализуется в виде дерева параллельных сумматоров (накопителей), служащего для сведения частичных произведений к вектору сумм и вектору переносов. Сжатие реализуется несколькими рядами сумматоров, причем каждый ряд вносит задержку, свойственна одному полному сумматору;
- ступень заключительного суммирования, где осуществляется сложение вектора сумм и вектора переносов с целью получения конечного результата.





Суммирование ЧП с помощью дерева Уоллеса : а — логика суммирования; б — точечная диаграмма

Как-Сорте со старших разрядов
 0.01000 8×6

$$\begin{array}{r} +0000000000000000 \\ 0000000100000 \\ \hline 00000100000 \\ \leftarrow 00000100000 \\ +111110000 \\ \hline 0...000110000 \end{array}$$

$$\begin{array}{r} 00110 \\ \leftarrow +2A \quad -2A \end{array}$$

$$\begin{array}{l} 0.01000 + 8 \\ 0.10000 + 2A \\ 1.10000 - 2A \end{array}$$

**последний
связь не выполняется*

0.0101×0.1101

5×13

$$\begin{array}{r} +0000000000 \\ 0...0010100 \\ \hline 0...00010100 \\ \leftarrow 00001010000 \\ +1111110110 \\ \hline 0.001000110 \\ +1.111111011 \\ \hline 0001000001 \end{array}$$

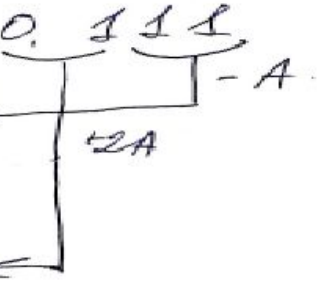
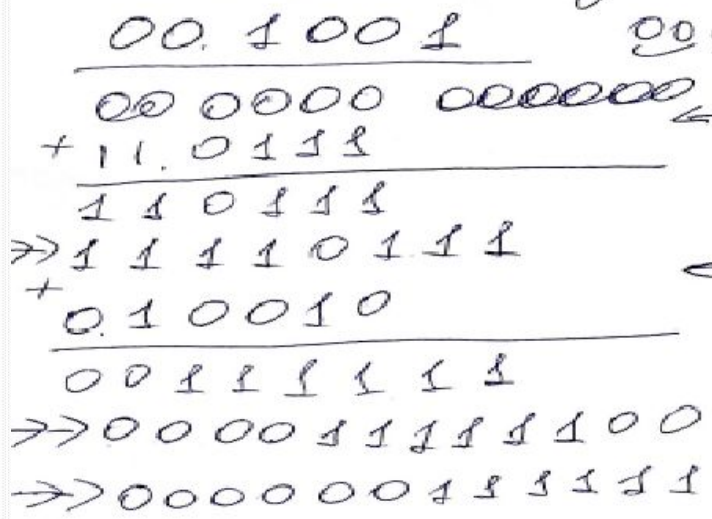
$$\begin{array}{r} 0.1101 \\ +4A \quad -2A \end{array}$$

$$\begin{array}{l} 0.0101 + 5 \\ 0.10100 + 4A \\ 1.10110 - 2A \\ 1.1011 - A \end{array}$$

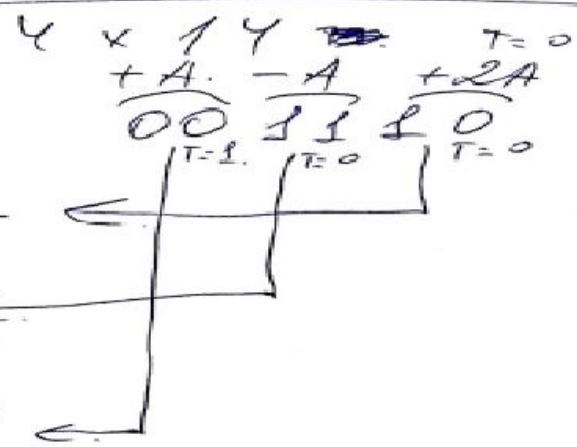
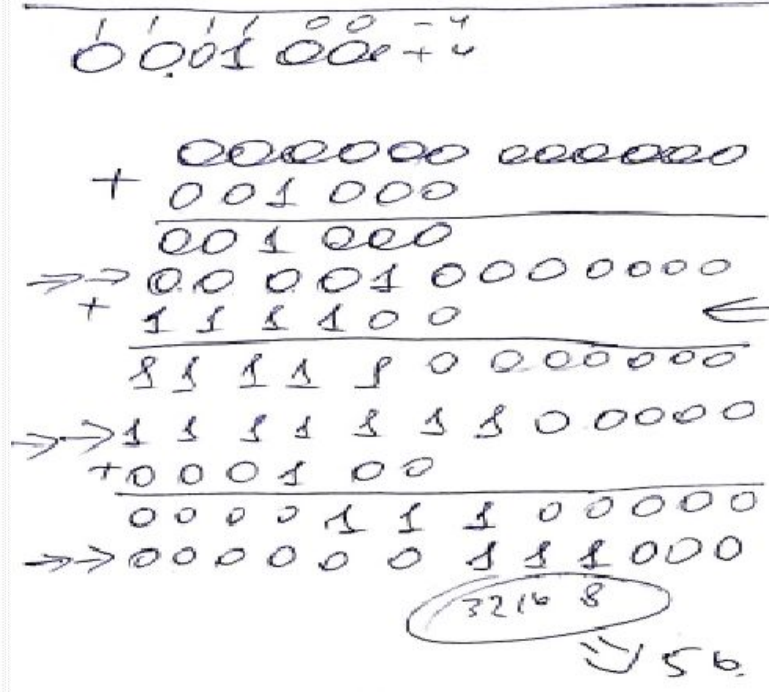
**корректируется*



Task - Copy with multiplexers



$-A = 110111$
 $+2A = 010010$



3216 8
 56

Arifmetika Bermana

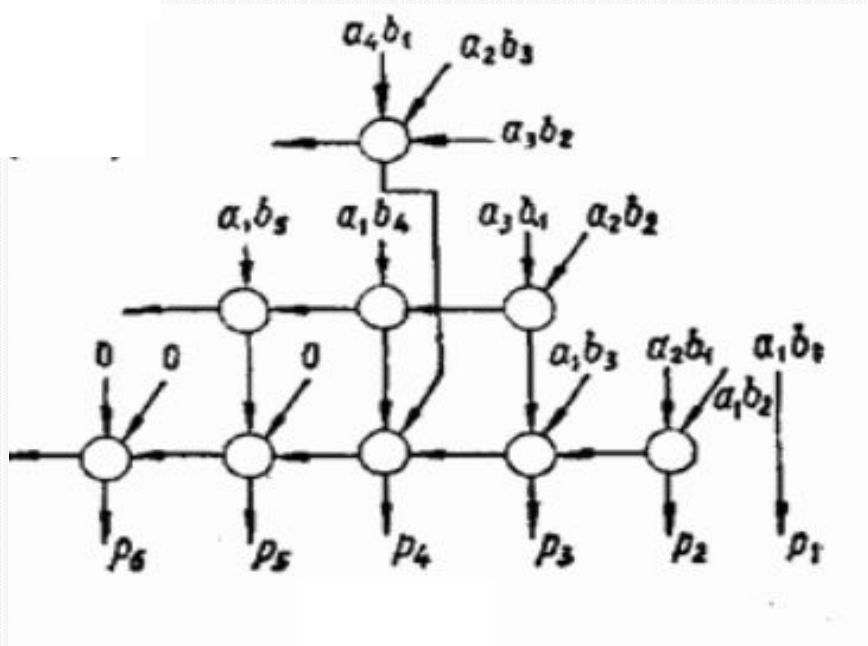
$5 \times 7 = 0.101 \times 0.111$

00000.00000	0.1	00	100	101
$+ 11011$				
<hr/>				
1101100000				
$\rightarrow 111011$				
1111101100				
$+ 00101$				
<hr/>				
0010001100				
$\rightarrow 0001000110$				
$\rightarrow 0000100011$				

0.00100	0.1101	4×13
$+ 000000000000$	0.10101	
$+ 00100$		
<hr/>		
000100000000		
$\rightarrow 00001000000$		
$\rightarrow 0000001000000$		
$+ 111100$		
<hr/>		
111101000000		
$\rightarrow 1111101000000$		
$\rightarrow 1111111010000$		
$+ 000100$		



Произведем распределение первичных и запаздывающих слагаемых



Для одноразрядных сумматоров время образования суммы больше, чем время образования переноса τ_n , поэтому наибольшее время спуска по дереву данного вида

$$T_y = (n - 1)(\tau_{см} + \tau_n),$$

$$2^j \leq m \leq 2^{j+1} - 1,$$

где m — число слагаемых. Время спуска по этому дереву составит

$$\tau_{сп} = j\tau_{см}.$$

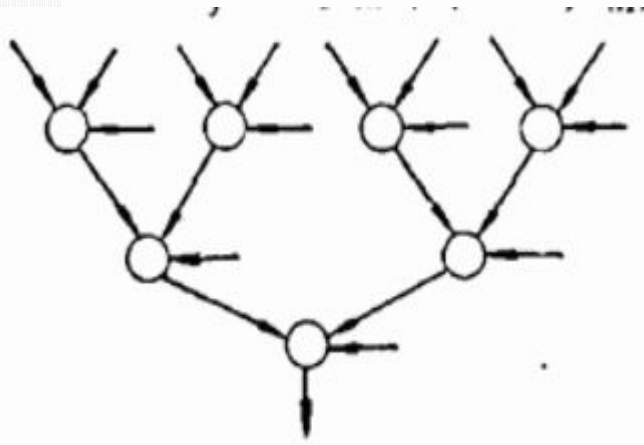


Рисунок 1

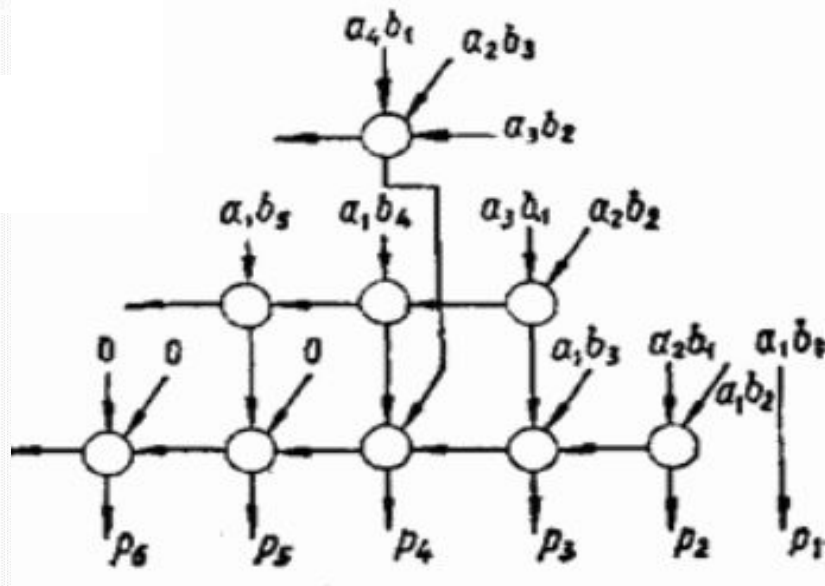
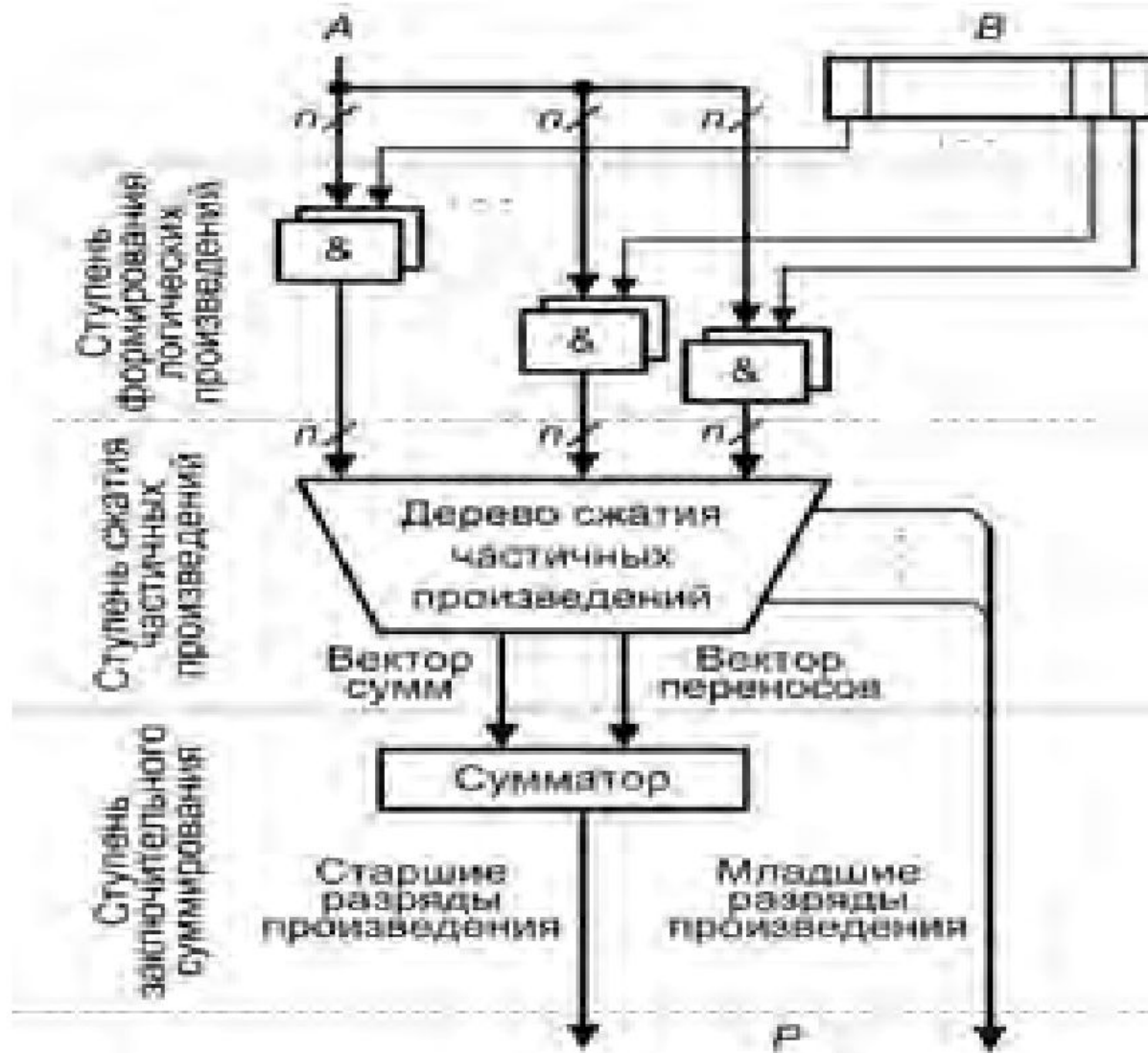


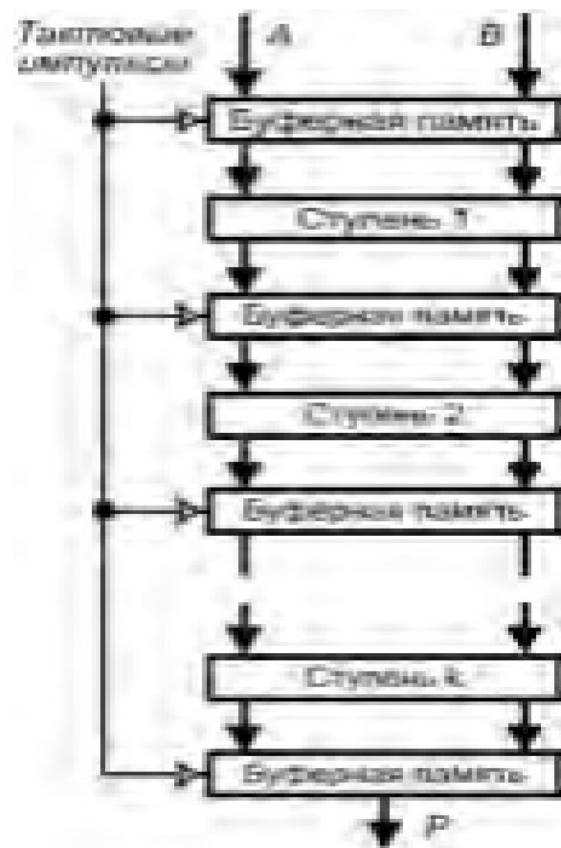
Рисунок 2, время спуска уменьшается до величины

$$T_y = 2[\tau_{см} + (n - 2)\tau_n],$$



Структура древовидного умножителя





Структура конвейерного умножителя

