



- Учебная дисциплина
- Схемотехника
- дискретных устройств
- Тема: Арифметические устройства

# Назначение и классификация арифметических устройств.

---

- Арифметические устройства предназначены для выполнения самых распространённых операций: сложения, вычитания, умножения и сравнения.

# Назначение и классификация арифметических устройств

---

- Основные устройства, выполняющие перечисленные операции, это сумматоры, вычитатели, множительные блоки и компараторы. Они входят в состав более сложных функциональных устройств вычислительных устройств, таких как, например, арифметико-логическое устройство (АЛУ).

# Назначение и классификация арифметических устройств

---

- Арифметическое сложение считается основной элементарной операцией, выполняемой над кодами чисел в цифровых устройствах.

# Назначение и классификация сумматоров.

---

- **Сумматор** - это электронное устройство, выполняющее арифметическое сложение кодов двух чисел.

# Определение сумматора

---

- Сумматоры применяются и для выполнения операции вычитания, но для этого осуществляются дополнительные преобразования кодов чисел.

# Классификация сумматоров

---

- Сумматоры классифицируют по различным признакам:

# Классификация сумматоров

---

- В зависимости от *системы счисления* различают:
- - *двоичные сумматоры;*
- - *двоично-десятичные;*
- - *десятичные;*
- - и другие.



# Классификация сумматоров

---

- По **количеству** одновременно обрабатываемых **разрядов** складываемых чисел сумматоры бывают:
  - - **одноразрядные;**
  - - **многоразрядные.**

# Классификация сумматоров

---

- По *числу входов и выходов* одноразрядных двоичных сумматоров различают:
  - - *четвертьсумматоры* (элементы "сумма по модулю 2", то есть "исключающее ИЛИ"), имеют два входа для двух одноразрядных чисел и одним выходом, на котором реализуется их арифметическая сумма;

# Классификация сумматоров

---

- - **полусумматоры**, характеризующиеся наличием двух входов, на которые подаются одноимённые разряды двух чисел и двух выходов: на одном реализуется **арифметическая сумма** в данном разряде, а на другом - **перенос** в следующий (более старший разряд);

# Классификация сумматоров

---

- - **полные одноразрядные** двоичные **сумматоры**, характеризующиеся наличием трёх входов, на которые подаются одноимённые разряды двух складываемых чисел и перенос из предыдущего (более младшего) разряда и двумя выходами:
  - на одном реализуется арифметическая сумма в данном разряде,
  - а на другом - перенос в следующий (более старший разряд).

# Классификация сумматоров

---

- По способу представления и обработки складываемых чисел многоразрядные сумматоры подразделяются на:
  - - **последовательные**, в которых обработка чисел ведётся поочерёдно, разряд за разрядом на одном и том же оборудовании;
  - - **параллельные**, в которых слагаемые складываются одновременно по всем разрядам, и для каждого разряда имеется своё оборудование.

# Классификация сумматоров

---

- По *способу выполнения операции сложения* и возможности сохранения результата выделяются два типа сумматоров:
  - - *комбинационный сумматор*, выполняющий микрооперацию " $S=A+B$ ", в котором результат выдаётся по мере его образования;
  - - *накапливающий сумматор*, на вход которого операнды подаются
- последовательно с некоторой задержкой.

# Классификация сумматоров

---

- По способу организации межразрядных переносов делят на:
- Сумматоры с **последовательным переносом**,
- Сумматоры с **параллельным переносом**,
- Сумматоры с **условным переносом**,
- Сумматоры со **сквозным переносом**

# Важнейшие параметры сумматоров

---

- Разрядность,
- Статические параметры:  $U_{вх.}$ ,  $U_{вых.}$
- $I_{вх.}$ , то есть обычные параметры интегральных схем.
- Динамические параметры:
- -задержка распространения от подачи входного переноса до установления всех выходов суммы при постоянном уровне на всех входах слагаемых,



# Важнейшие параметры сумматоров

---

- - задержка распространения от одновременной подачи всех слагаемых до установления всех выходов суммы при постоянном уровне на входе переноса,
- - задержка распространения от подачи входного переноса до установления выходного переноса при постоянном уровне на входах слагаемых,

# Важнейшие параметры сумматоров

---

- - задержка распространения от подачи всех слагаемых до установления выходного переноса при постоянном уровне на входах слагаемых.

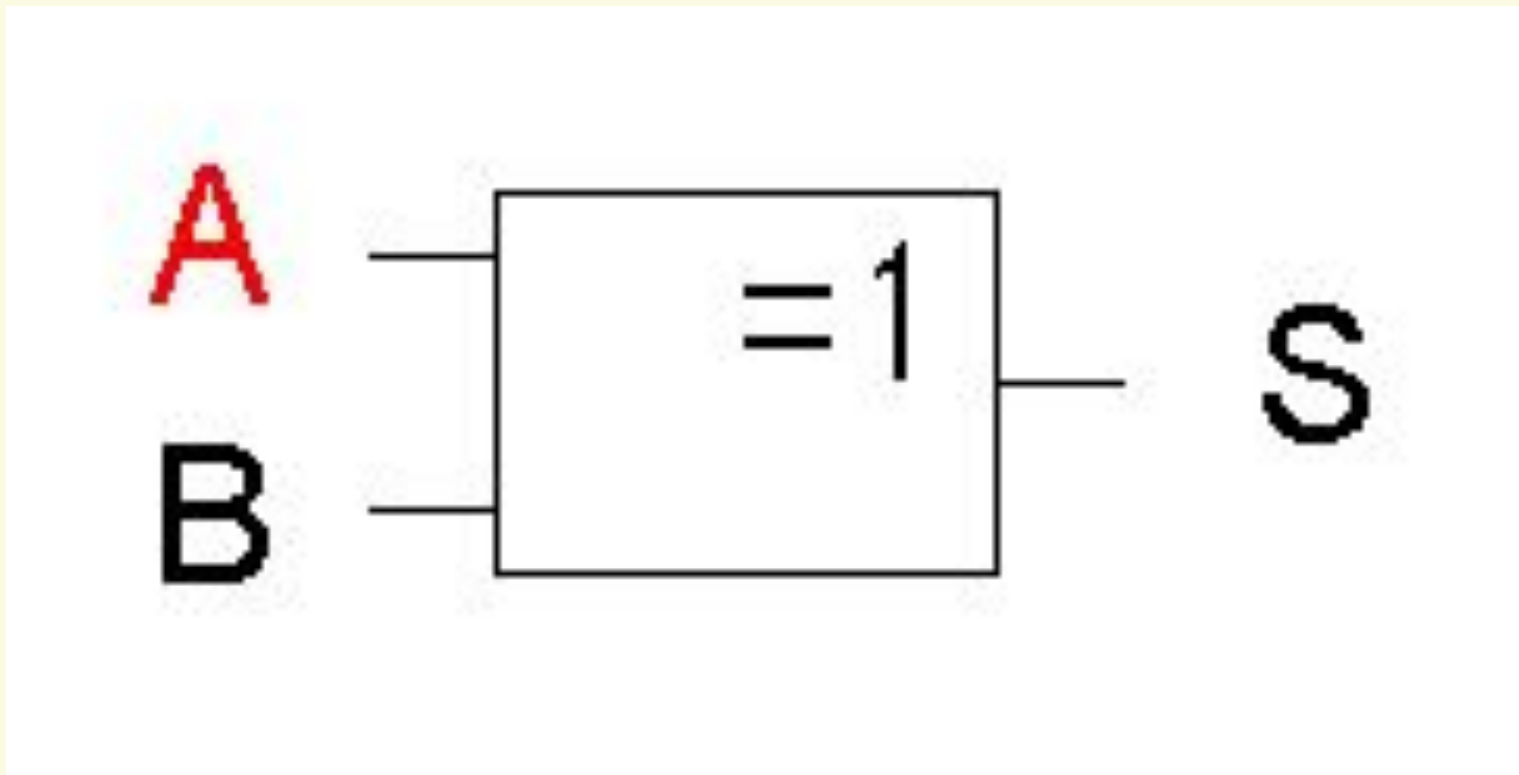
# Четвертьсумматор

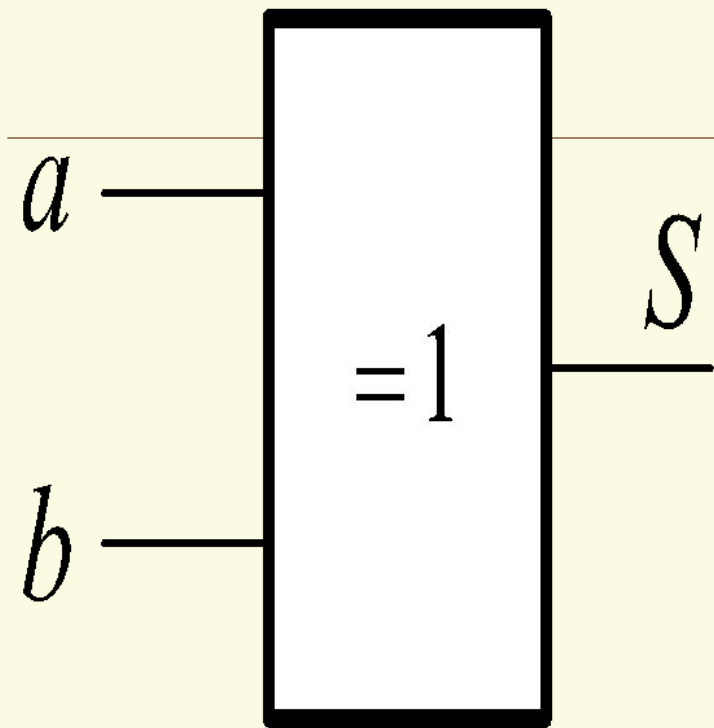
---

- Простейшем двоичным суммирующим элементом является ***Четвертьсумматор***. Он реализуется логическим элементом «исключающее ИЛИ». Схема имеет два входа  $A$  и  $B$  и один выход суммы  $S$ .

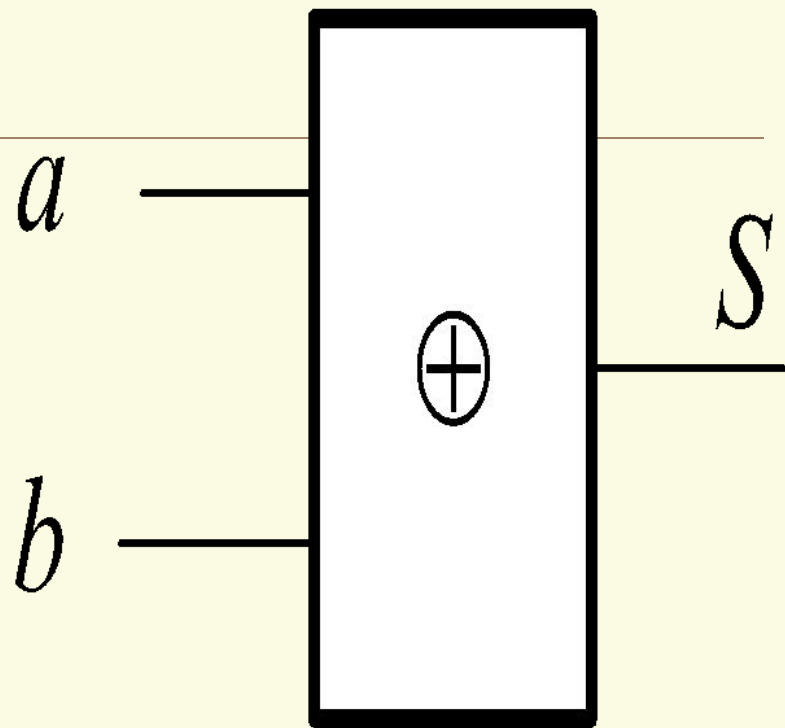
# Четвертьсумматор

- $S = \bar{a}b + a\bar{b}$





$a)$



$b)$

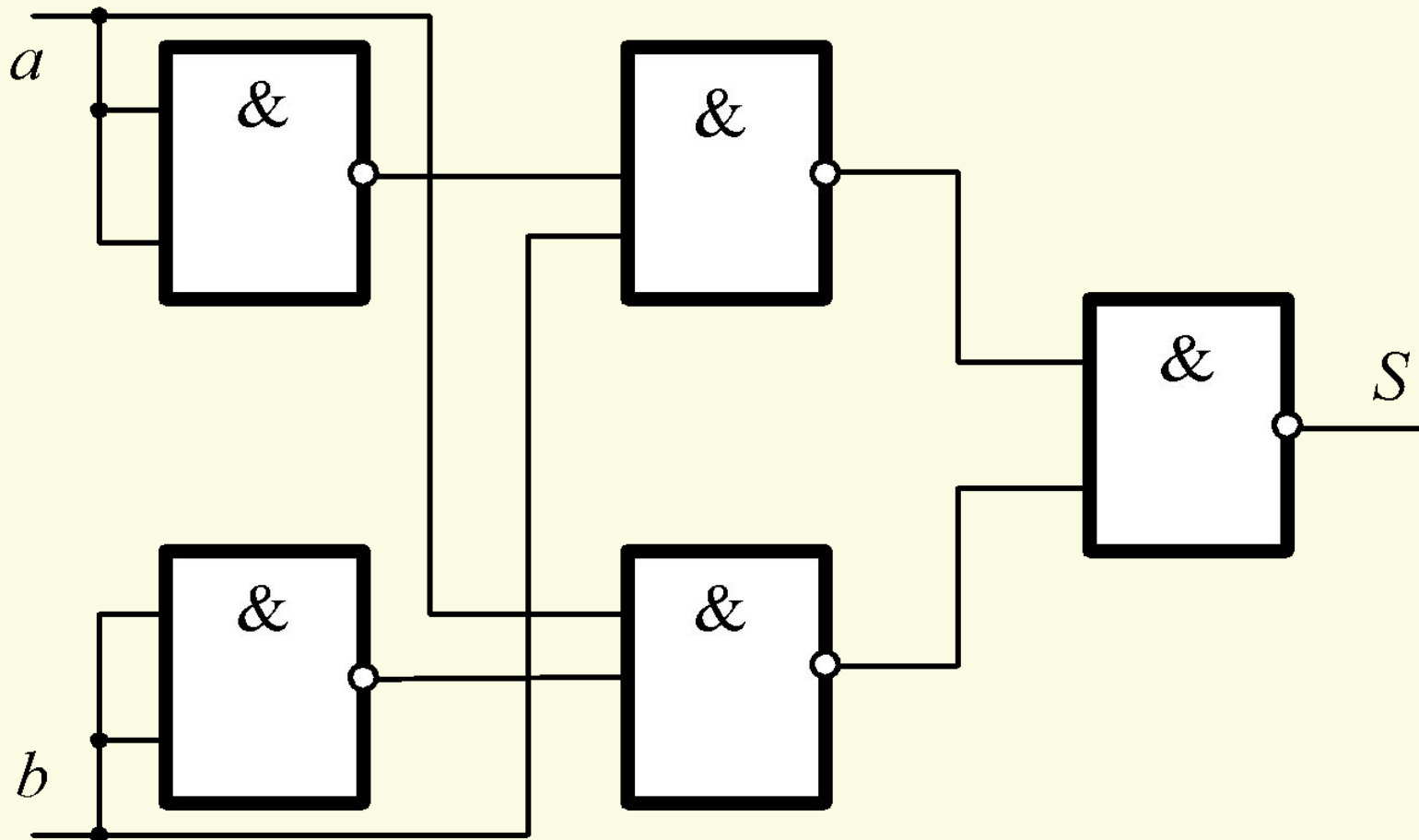
# Таблица функционирования четвертьсумматора

a	b	S
0	0	0
0	1	1
1	0	1
1	1	0

- По таблице истинности можно составить булеву функцию для суммы .

$$S = \overline{a}b + a\overline{b} = \overline{\overline{\overline{a}b} \overline{a\overline{b}}}$$

# схема четвертьсумматора на элементах «И-НЕ»





# Полусумматор

---

- ***Полусумматор*** - это комбинационная схема, которая вырабатывает сигналы суммы (S) и переноса (C) при сложении двух двоичных чисел  $a$  и  $b$ .
- Составим таблицу функционирования:

# Полусумматор

<b>a</b>	<b>b</b>	<b>S</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>
<b>1</b>	<b>0</b>	<b>1</b>	<b>0</b>
<b>0</b>	<b>1</b>	<b>1</b>	<b>0</b>
<b>1</b>	<b>1</b>	<b>0</b>	<b>1</b>

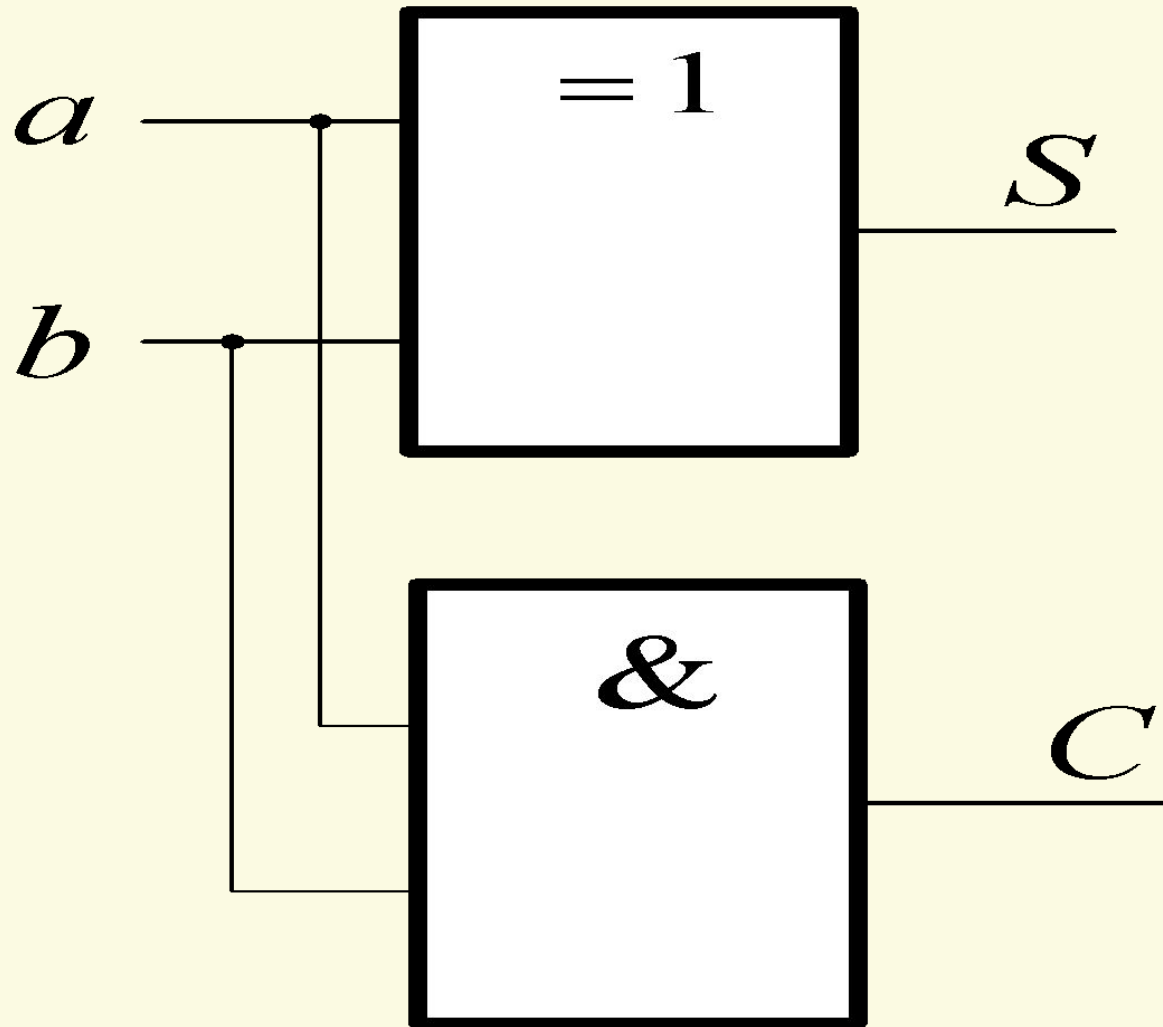
- 
- Из таблицы следует, что функция суммы реализуется логическим элементом «Исключающее ИЛИ», а функция переноса реализуется логическим элементом «И».

# Полусумматор

---

- Из таблицы получим:
- $S = a \oplus b + \bar{a}b$  – сигнал суммы;
- $C = ab$  – сигнал переноса.
- Эти выражения упрощению не поддаются.
- Приведем к виду, удобному для реализации на элементах ИЛИ-НЕ.

# Типовая схема полусумматора на элементах «Исключающее ИЛИ» и «И»



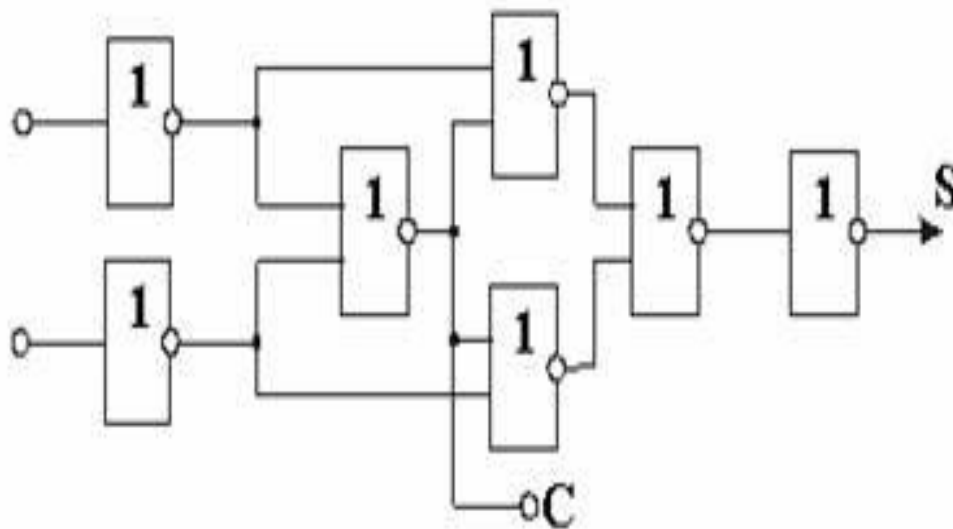
# Полусумматор

$$S = a\bar{b} + \bar{a}b = a(\bar{b} + \bar{a}) + b(\bar{a} + \bar{b}) = \overline{\overline{a(\bar{b} + \bar{a})}} + \overline{\overline{b(\bar{a} + \bar{b})}} =$$
$$\overline{\overline{a(\bar{b} + \bar{a})}} + \overline{\overline{b(\bar{a} + \bar{b})}} = \overline{\overline{a}} + \overline{\overline{b}} + \overline{\overline{a}} + \overline{\overline{b}} = \overline{\overline{a}} + \overline{\overline{b}}.$$

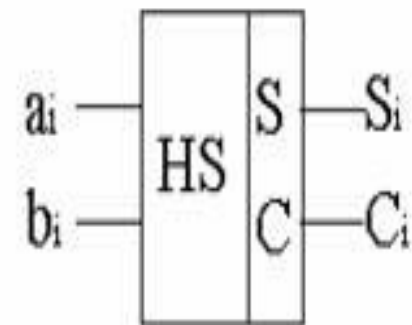
$$C = ab = \overline{\overline{ab}} = \overline{\overline{a}} + \overline{\overline{b}}$$

# Схема полусумматора

- Исходя из полученных формул составим схему полусумматора



a)



б)

# Синтез полусумматора

---

- Составляя дизъюнктивную нормальную форму для полусумматора, мы получили следующие булевы функции:

$$C = ab \quad \text{и} \quad S = \bar{a}b + a\bar{b} = a \oplus b.$$



# Синтез полусумматора

---

- Следовательно, перенос происходит с помощью **функции И**, а выработка сигнала суммы (функции неравнозначности) производится элементом **ИСКЛЮЧАЮЩЕЕ – ИЛИ**. На след. слайде показана схема полусумматора, составленная из этих элементов.

# Сумматор (полный сумматор)

---

- В отличие от *полусумматора* должен воспринимать 3 входных сигнала: **2 слагаемых и сигнал переноса с предыдущего разряда.**
- **Сумматором называется операционный узел ЭВМ, выполняющий операцию арифметического сложения двух чисел.**
- Чтобы понять сущность работы комбинационного сумматора, рассмотрим примеры суммирования двух одnorазрядных двоичных чисел:

# Сумматор

$$1) \begin{array}{r} + 1 \\ 0 \\ \hline 1 \end{array}$$

$$2) \begin{array}{r} + 0 \\ 1 \\ \hline 1 \end{array}$$

перенос в ст. разряд

$$3) \begin{array}{r} \overset{1}{\curvearrowright} \\ + 1 \\ \hline 0 \end{array}$$

перенос в ст. разряд

перенос из мл. разряда

$$4) \begin{array}{r} \overset{1}{\curvearrowright} \quad \overset{1}{\curvearrowright} \\ + 1 \\ \hline 1 \end{array}$$

# Сумматор

---

- Из приведенных примеров видно, что если отсутствует перенос из младшего разряда, то перенос в старший разряд может быть только в одном случае, когда оба числа равны единице.
- Если же имеется перенос из младшего разряда, то перенос в старший разряд будет всегда, кроме одного случая, когда оба слагаемых равны нулю.

# Таблица функционирования сумматора

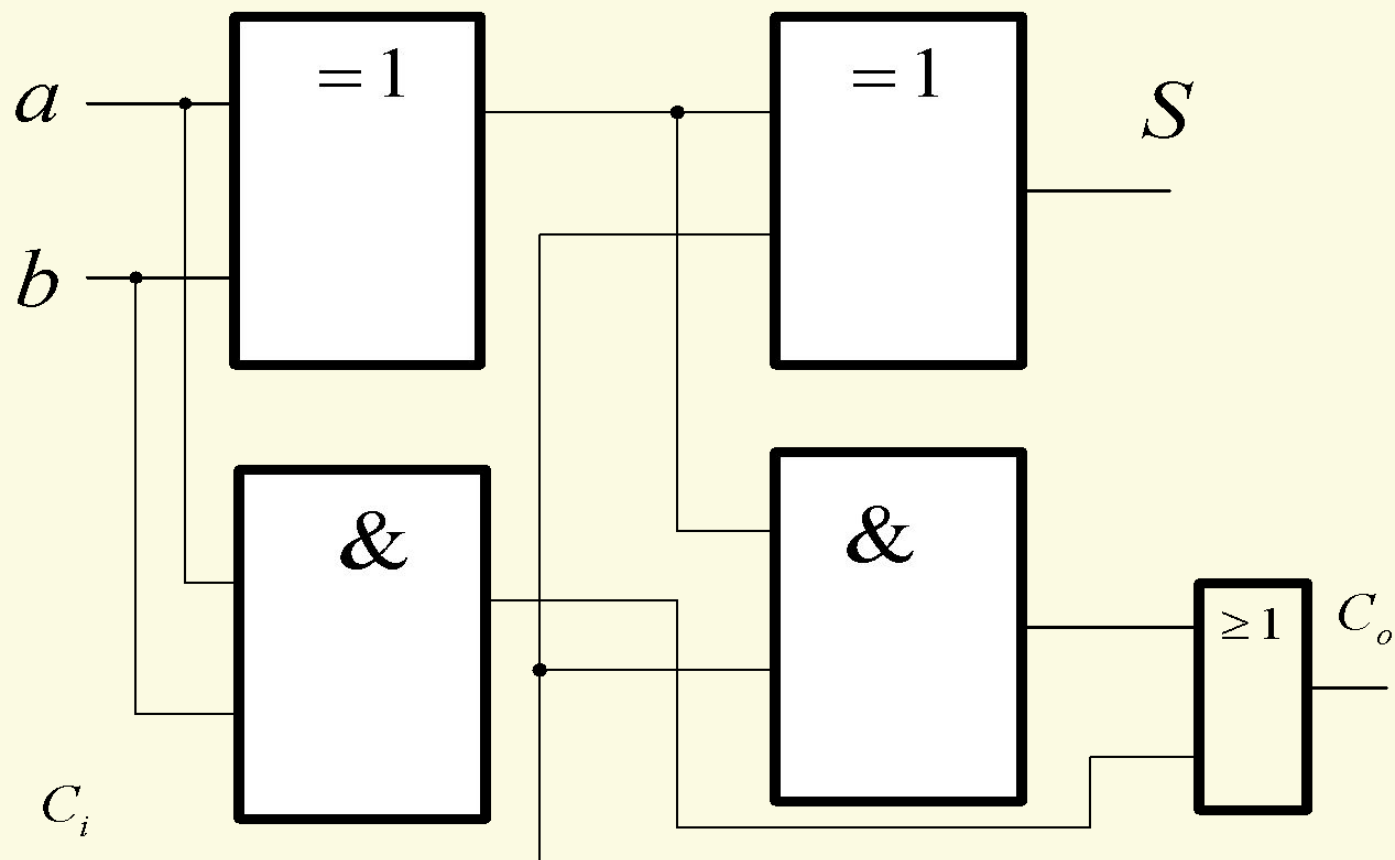
$a_i$	$b_i$	$C_i$	$S_i$	$C_{i+1}$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

# Сумматор из двух полусумматоров

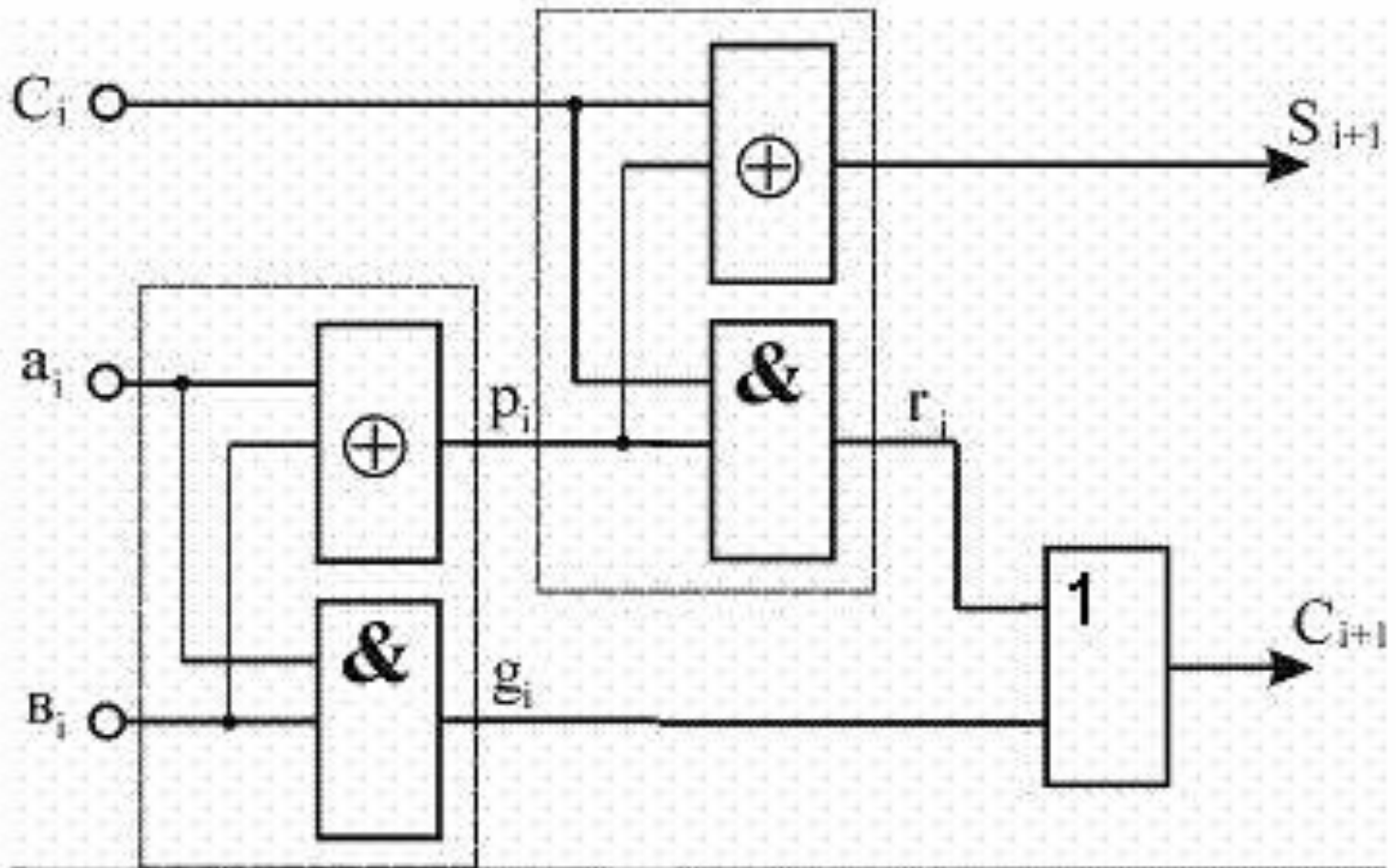
---

- Схема сумматора может быть реализована ***на двух полусумматорах.***
- В этой схеме выделим промежуточные сигналы  $P_i$ ,  $g_i$ ,  $r_i$ . Введем эти сигналы в новую таблицу функционирования. Соответствие работы этой схемы и таблицы функционирования можно проверить перебором всех возможных вариантов.

# Схема полного сумматора из двух полусумматоров.



# Схема полного сумматора

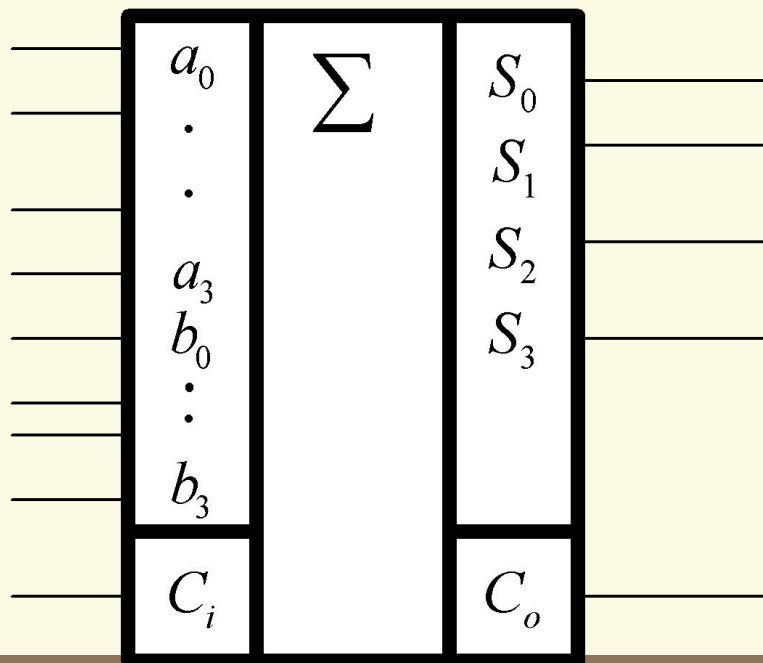




# Таблица функционирования полного сумматора

Входы			Промежуточные величины			Выходы	
$a_i$	$b_i$	$c_i$	$P_i$	$g_i$	$r_i$	$S_i$	$C_{i+1}$
0	0	0	0	0	0	0	0
0	1	0	1	0	0	1	0
1	0	0	1	0	0	1	0
1	1	0	0	1	0	0	1
0	0	1	0	0	0	1	0
0	1	1	1	0	1	0	1
1	0	1	1	0	1	0	1
1	1	1	0	1	0	1	1

- Согласно ГОСТ 2.743-91 условно-графическое изображение сумматора выглядит следующим образом

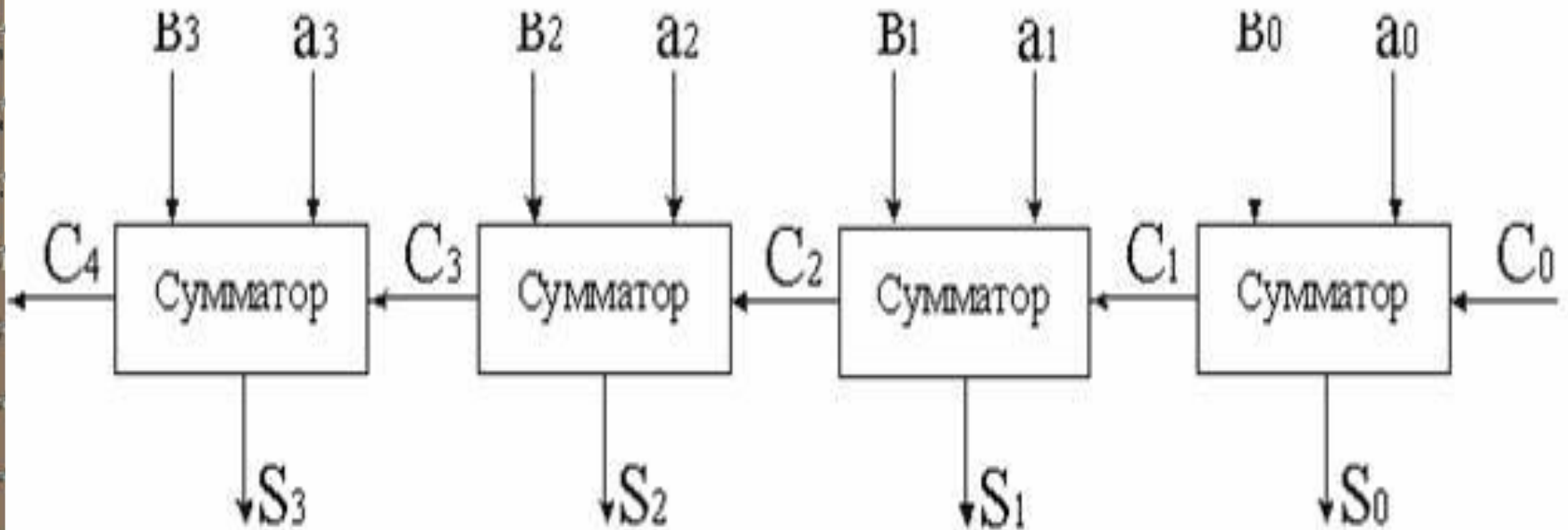


# Многоразрядный сумматор с последовательным переносом.

---

- Для сложения двух многоразрядных двоичных чисел на каждый разряд необходим один полный сумматор. Только в младшем разряде можно обойтись полусумматором.
- На след. слайде приведена схема, предназначенная для сложения двух четырехразрядных чисел  $A$  и  $B$ .

# Многоразрядный сумматор с последовательным переносом.

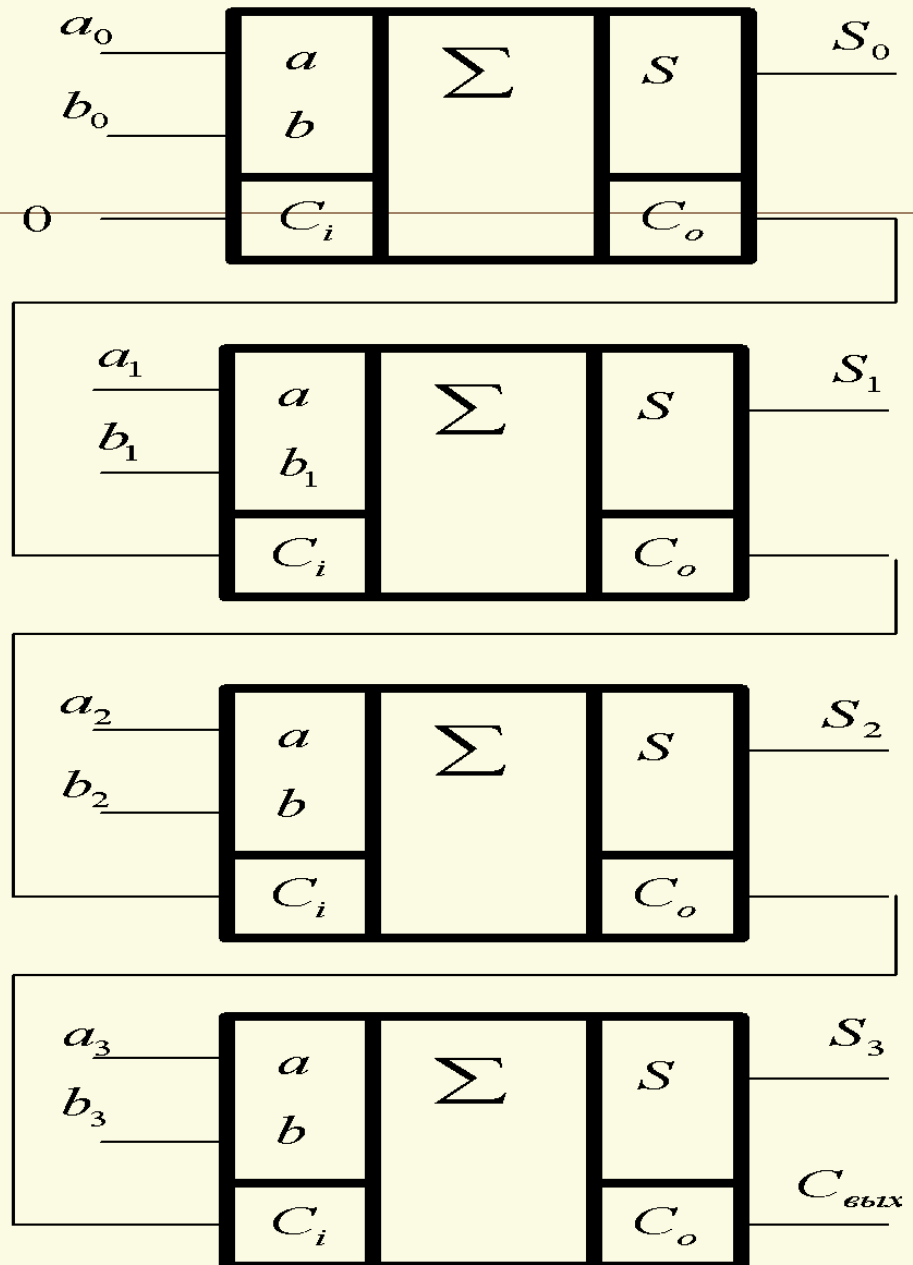


# Сумматор параллельного действия

---

- В сумматоре параллельного действия аргументы подаются одновременно по всем разрядам.
- Состоит из отдельных разрядов, каждый из которых содержит одноразрядный сумматор

# Сумматор параллельного действия



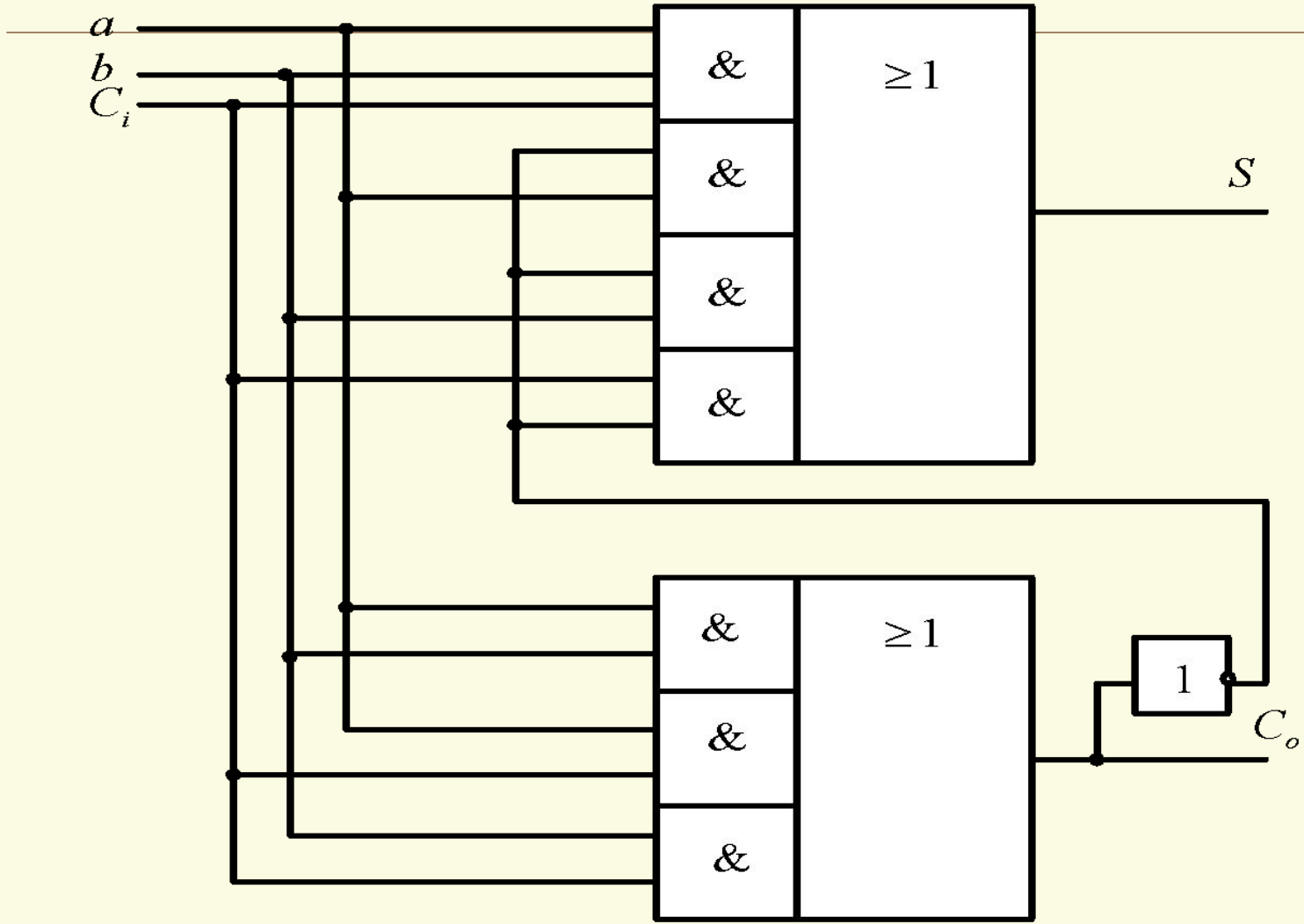
- 
- Одноразрядный комбинационный сумматор можно реализовать и в другом базисе, например на логических элементах основного функционально-полного набора («И-ИЛИ-НЕ»). Для этого необходимо сделать синтез схемы для функции суммы и функции переноса .

- Результатом синтеза являются характеристические выражения для суммы и выходного переноса.

$$S = C_i \overline{C_o} + abC_i + aC_o + b\overline{C_o} = abC_i + C_i + 1(a + b + C_i)$$

$$C_o = \overline{a}bC_i + a\overline{b}C_i + ab\overline{C_i} + abC_i = ab + C_i(a + b)$$





# Сумматор последовательного действия

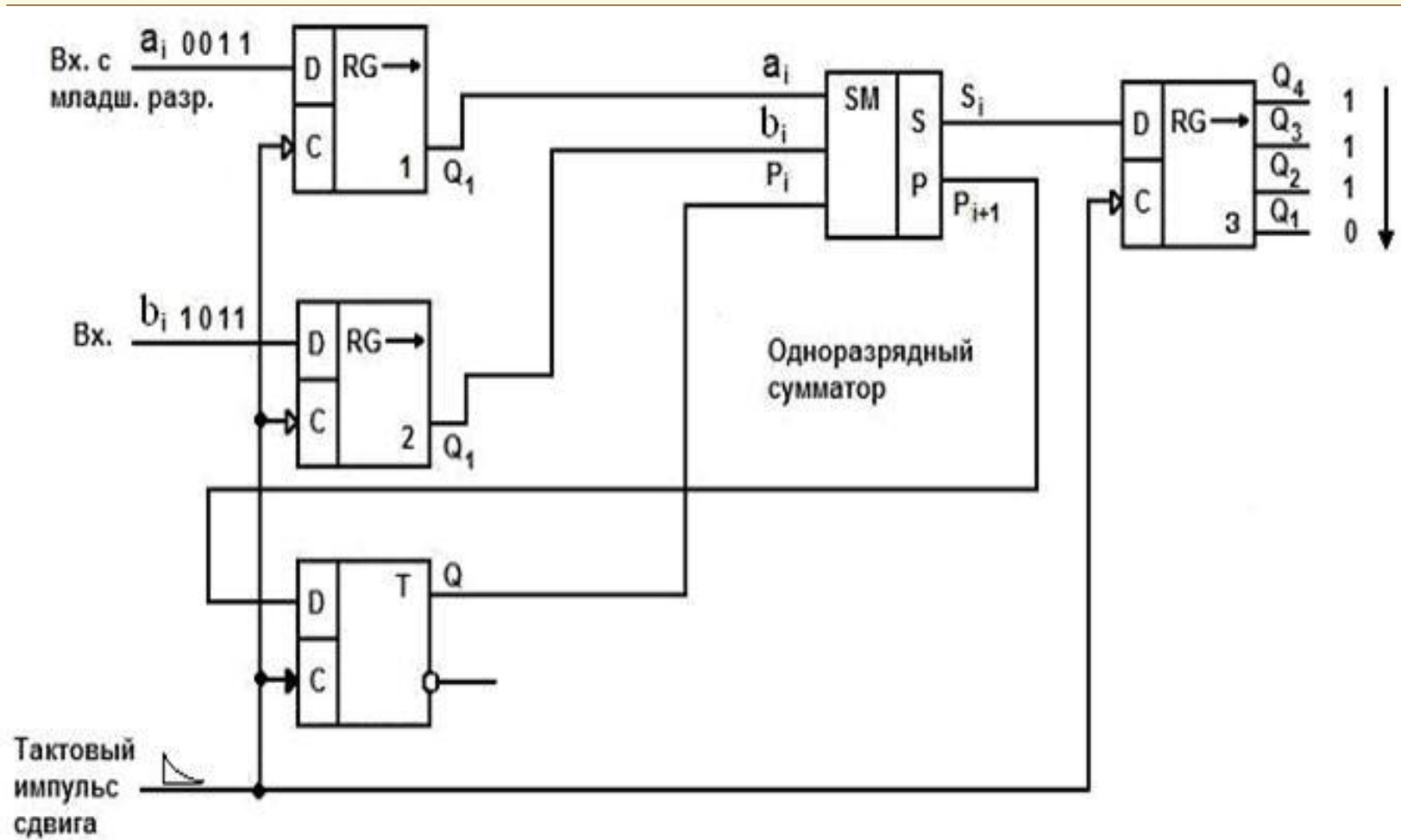
- Сумматор последовательного действия. Состоит из одноразрядного сумматора, выход  $r_i+1$  которого соединен с входом  $r_i$  через элемент задержки, параметры которого согласованы со скоростью поступления разрядов слагаемых на входы сумматора. Операция суммирования во всех разрядах слагаемых осуществляется с помощью одного и того же одноразрядного сумматора, но последовательно во времени, начиная с младших разрядов. Такое построение сумматора возможно за счет того, что слагаемые поступают в последовательной форме.

# Сумматор последовательного действия

---

- Очевидное достоинство **сумматора последовательного действия** заключается в малом объеме оборудования, требуемого для его построения. Однако связанная с этим необходимость в последовательной обработке разрядов приводит к крайне низкому быстродействию. Поэтому, сумматоры такого типа в настоящее время используются очень редко.

# Структура сумматора последовательного действия



- 
- Аргументы, участвующие в сложении, загружаются в два сдвигающих регистра.
  - Младшие разряды регистров соединены с одноразрядными входами сумматора.

- 
- Выход суммы одноразрядного сумматора соединён со входом сдвигающего регистра накопления суммы. Для сохранения возникающего переноса используется D – триггер , подающий перенос на сумматор при суммировании следующих разрядов.

- 
- Входные сдвигающие регистры и регистр суммы управляются двумя сдвинутыми относительно друг друга сериями синхроимпульсов, учитывающих время срабатывания одноразрядного сумматора.

# Организация переноса в многоразрядных сумматорах

---

- **Методы ускорения распространения переносов в сумматорах.**



# Недостаток сумматоров с последовательным переносом.

---

- Время выполнения операции в сумматоре с последовательным переносом намного больше времени сложения в одноразрядном сумматоре. Действительно, сигнал переноса  $C_4$  только тогда может принять истинное значение, когда будет установлено правильное значение  $C_3$ . Такой порядок выполнения операций называется последовательным переносом (Ripple Carry).

- Задержку распространения переноса можно определить из выражения:

$$T_c = t_{ic} * (n - 1)$$

- где  $t_{ic}$  - задержка появления переноса на одноразрядном сумматоре,  $n$  - количество разрядов сумматора.

## Сумматор параллельного действия с параллельным переносом

- При подаче слагаемых цифры их разрядов поступают на соответствующие одноразрядные сумматоры. Каждый из одноразрядных сумматоров формирует на своих выходах цифру соответствующего разряда суммы и перенос, передаваемый на вход одноразрядного сумматора следующего (более старшего) разряда. Такая организация процесса организации переноса, называемая последовательным переносом, снижает быстродействие многоразрядных сумматоров, т.к. получение результата в старшем разряде сумматора обеспечивается только после завершения распространения переноса по всем разрядам.

# Сумматоры с параллельным переносом.

---

- Чтобы уменьшить время операции сложения многоразрядных чисел можно использовать схемы параллельного переноса (Carry look-ahead). При этом все сигналы переноса вычисляются непосредственно по значениям входных переменных.
- Согласно таблице переключений, в общем случае для сигнала переноса любого  $i$ -го разряда справедливо соотношение:

# Сумматоры с параллельным переносом.

---

$$C_i = a_i b_i + (a_i \oplus b_i) C_i = g_i + P_i C_i \quad .$$

- Величины  $g_i$ ,  $r_i$  вычисляются в качестве промежуточных результатов и в полном сумматоре.

# Сумматоры с параллельным переносом.

---

- Следовательно, их получение не требует дополнительных затрат. Смысл этих величин объясняется совсем просто. Сигнал  $g_i$  вырабатывается тогда, когда в данном разряде перенос происходит из-за комбинации входных переменных  $a_i, b_i$ . **Поэтому его называют функцией генерации переноса.** Сигнал  $P_i$  показывает, передается ли полученный в младшем разряде сигнал переноса  $C_i$  дальше. Поэтому он называется **функцией распространения переноса или функцией транзита.**

# Сумматоры с параллельным переносом.

- Пользуясь выражением для  $C_i$ , можно вывести следующие формулы для вычисления сигналов переноса:

$$C_1 = g_0 + P_0 C_0,$$

$$C_2 = g_1 + P_1 C_1 = g_1 + P_1 g_0 + P_1 P_0 C_0,$$

$$C_3 = g_2 + P_2 C_2 = g_2 + P_2 g_1 + P_2 P_1 g_0 + P_2 P_1 P_0 C_0,$$

$$C_4 = g_3 + P_3 C_3 = g_3 + P_3 g_2 + P_3 P_2 g_1 + P_3 P_2 P_1 g_0 + P_3 P_2 P_1 P_0 C_0$$

# Сумматоры с параллельным переносом.

---

- Очевидно, что хотя полученные выражения достаточно сложные, время формирования сигнала переноса в любой разряд с помощью вспомогательных функций определяется только временем задержки распространения сигнала на двух элементах.
- Эти функции реализуются специальным комбинационным устройством – ***схемой ускоренного переноса.***



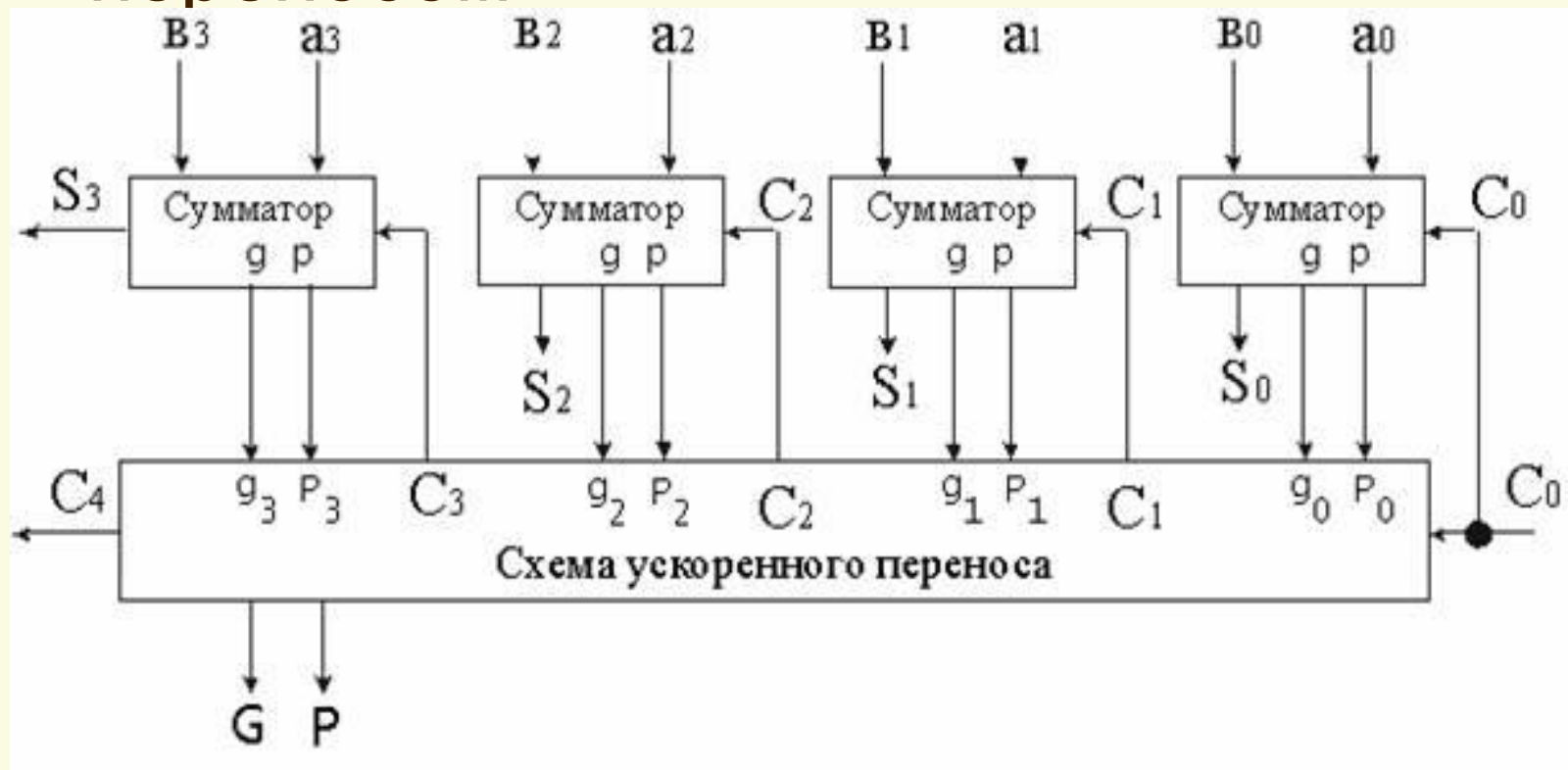
# Сумматоры с параллельным переносом.

---

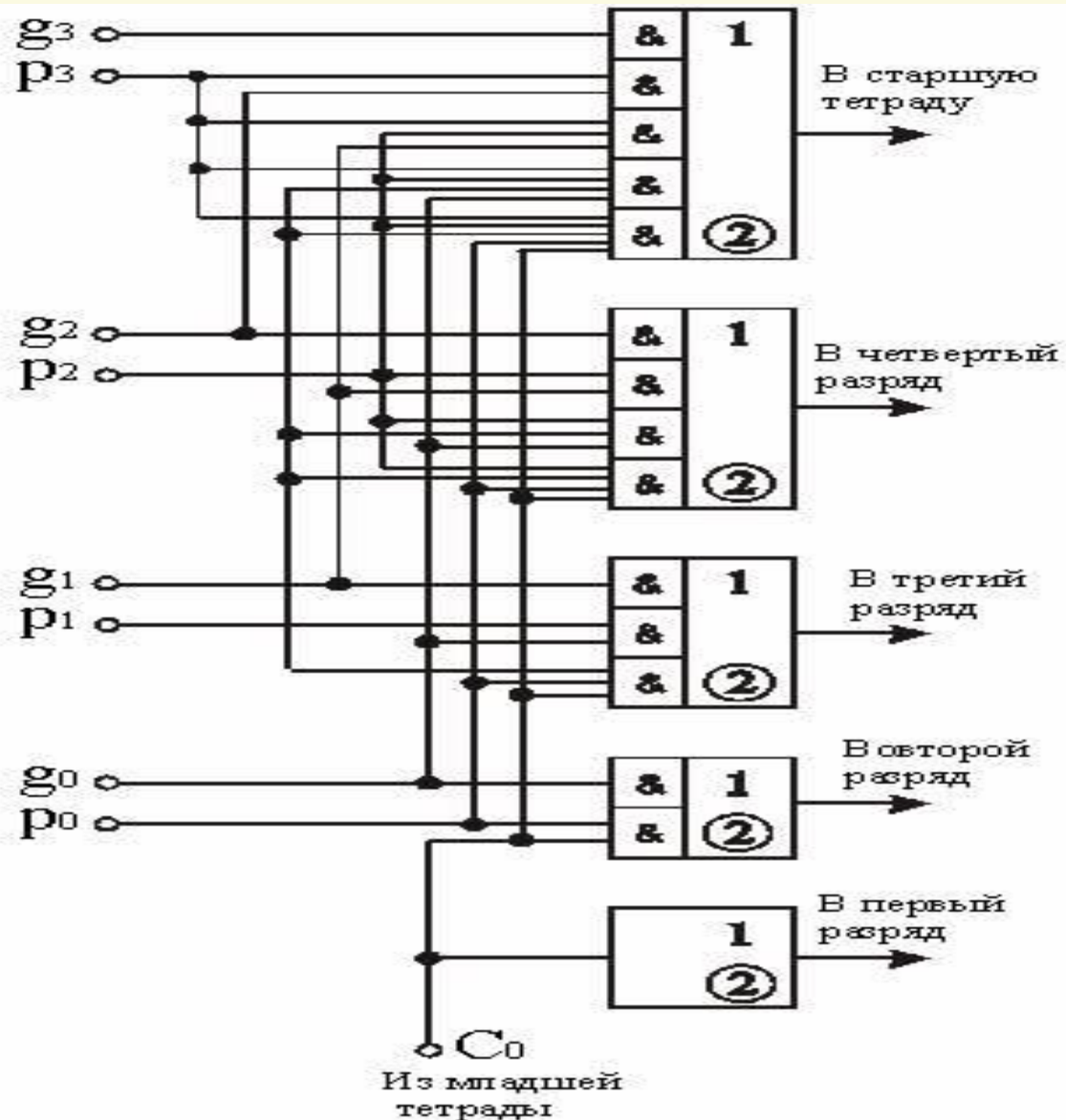
- Схема сумматора с параллельным переносом приведена на следующих слайдах и далее изображена схема устройства параллельного переноса в группе из четырех разрядов.

# Сумматоры с параллельным переносом.

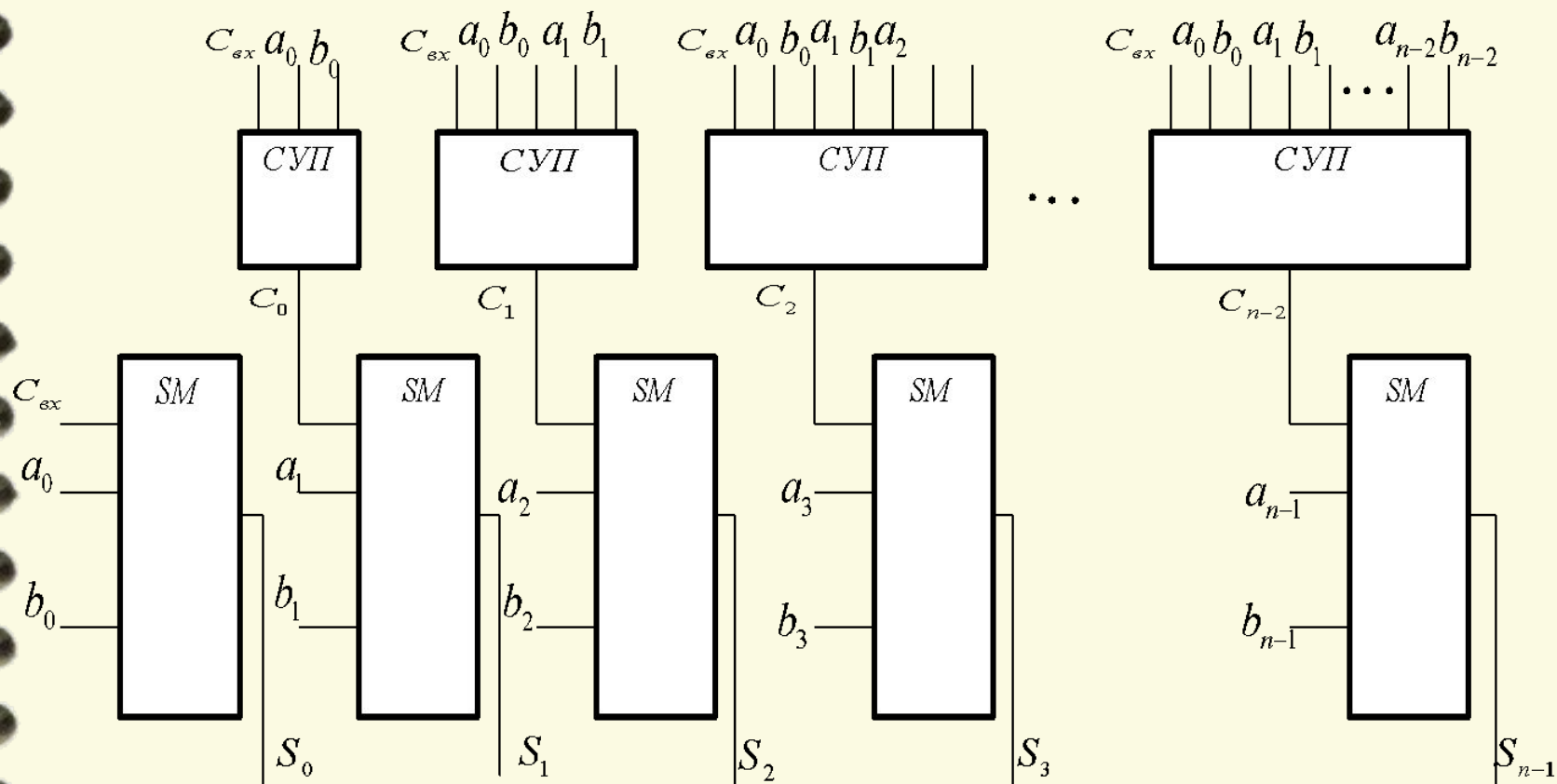
- Схема сумматора с параллельным переносом



# Схема ускоренного переноса



# Сумматор с параллельным переносом



# Оценка времени суммирования

---

- Время суммирования складывается из времени формирования функции прозрачности (одна задержка элемента И-НЕ, которую обозначим  $T_{ла}$ ), времени формирования функции переноса  $2T_{ла}$  и задержки одноразрядных сумматоров, что в результате даёт время суммирования  $= (4...5) T_{ла}$

# Оценка времени суммирования

---

- Длительность суммирования, полученная из рассмотрения логической схемы сумматора, не зависит от его разрядности, что является характерным признаком структур с параллельными переносами вообще, и не только сумматоров.

# Оценка времени суммирования

---

- Однако фактически это не совсем так, поскольку с ростом разрядности сумматора увеличивается нагрузка элементов схемы, что увеличивает их задержки. В частности, коэффициент разветвления элементов, вырабатывающих функции прозрачности, равен  $n^2/4$ , т.е. квадратично зависит от разрядности сумматора.

# Сумматоры с групповым переносом

---

- Схемы *группового переноса* применяют у сумматоров большой разрядности.
- Сумматор разбивается на группы с разрядностью от 2 до 8.
- Тракт межгруппового переноса может быть построен как последовательный, так и параллельный.



# Сумматоры с групповым переносом

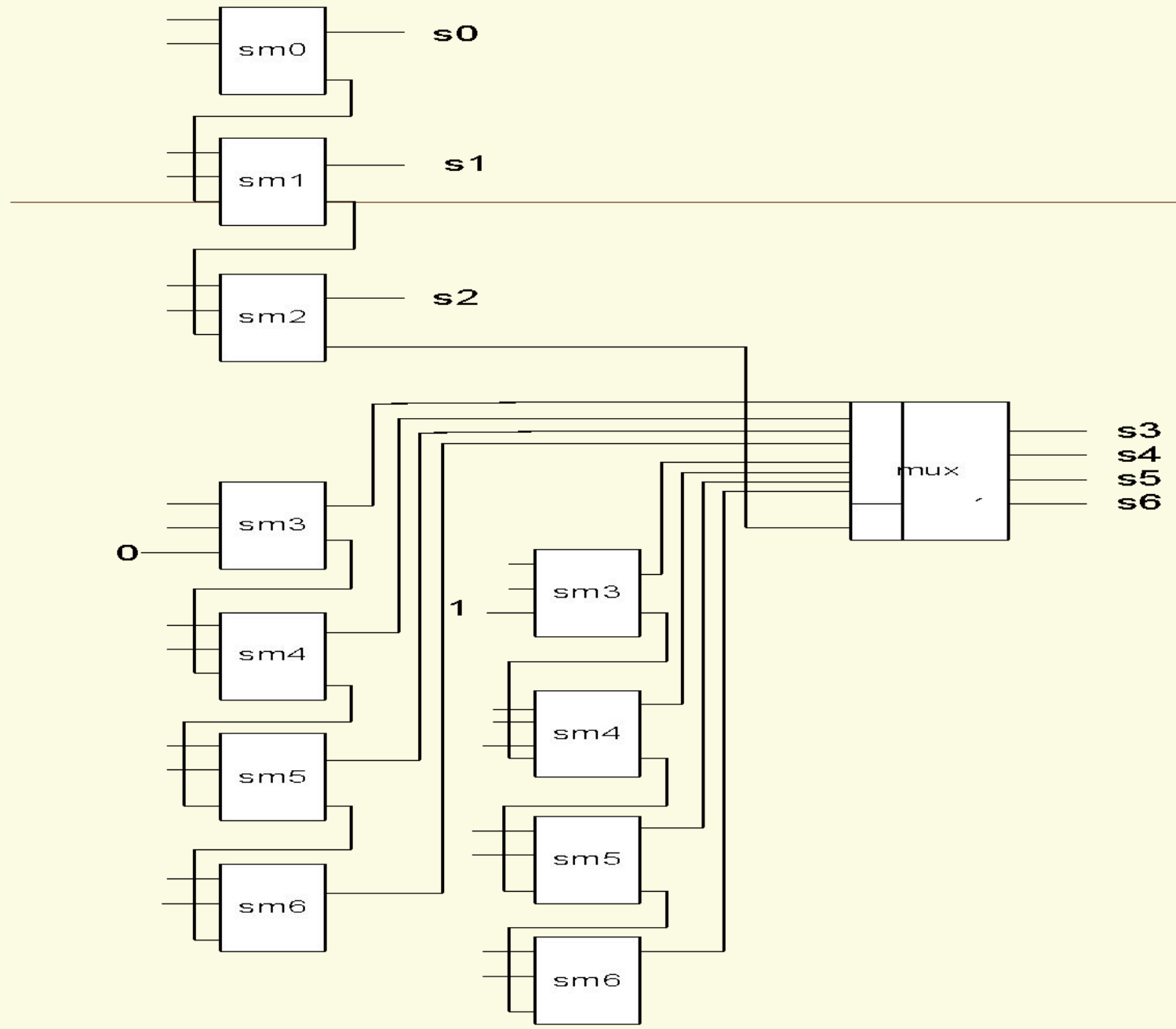
---

- Параллельный перенос между группами в сочетании с параллельным переносом внутри групп даёт самые быстрые сумматоры в диапазоне разрядности от 24 до 64-х.
- В диапазоне разрядности от 8 до 24-х первенство по скорости переходит к сумматорам с параллельным переносом между группами и последовательным внутри групп. Разрядность групп выбирают от 2 до 4-х.

# Сумматоры с условным переносом

---

- Сумматор разрядности  $n$  делят пополам на младшую и старшую группу разрядов. Старшую группу дублируют. В итоге сумматор состоит из трёх групп разрядов размерностью  $n/2$ . На одной старшей группе суммируются разряды при условии, что перенос из младшей группы отсутствовал, а на другой старшей, что перенос был. При завершении операции на младших разрядах становится ясным какой результат использовать на старших группах. Выбором старшей группы управляют мультиплексоры, управляемые переносом из младшей группы.





# Двоично-десятичные сумматоры

---

- Данные сумматоры выполняют действия над десятичными числами, разряды которых заданы двоичными тетрадами.
- Способ построения данного сумматора предусматривает первичное суммирование тетрад обычным двоичным сумматором и последующую коррекцию результата.

# Двоично-десятичные сумматоры

---

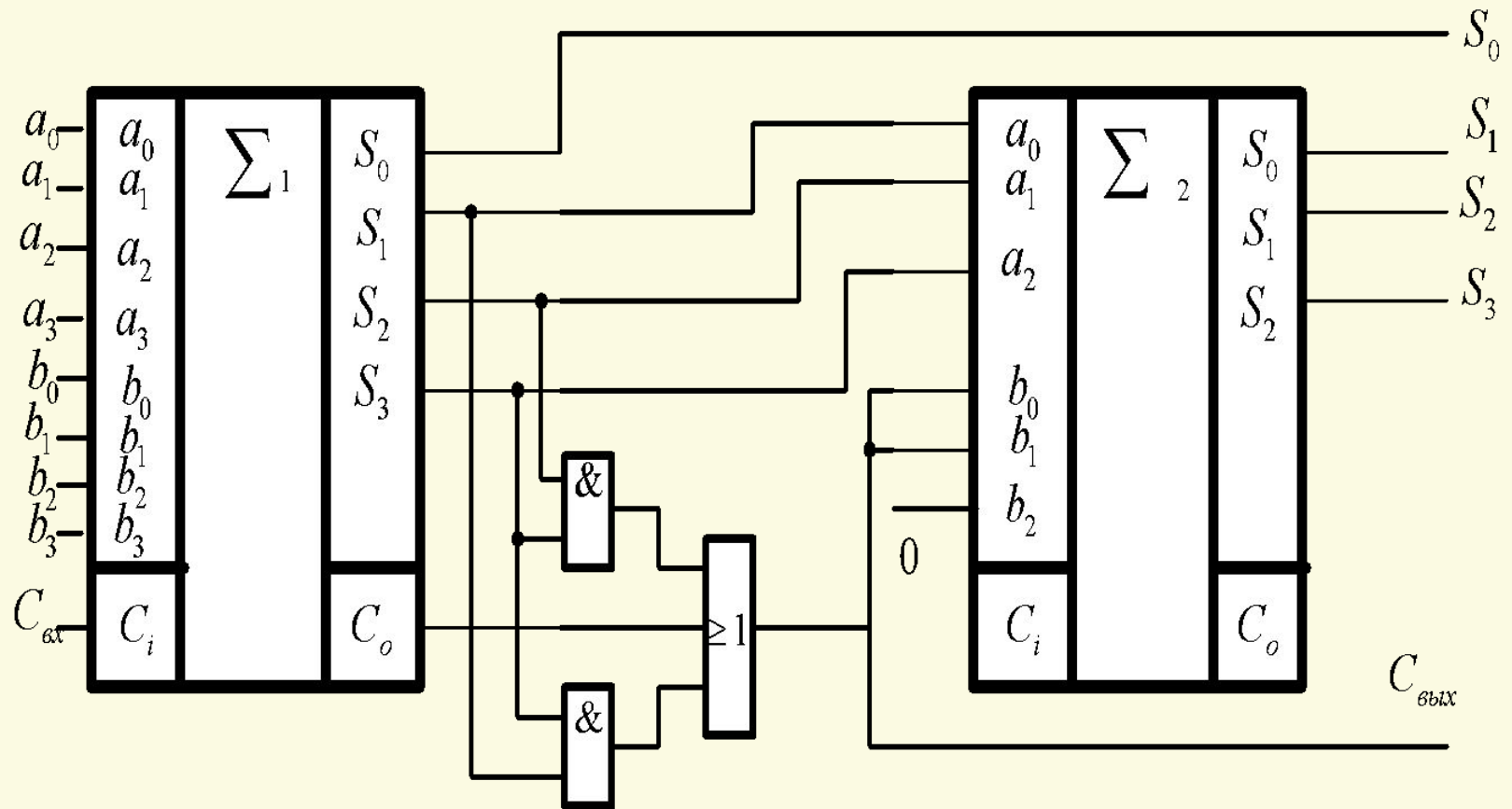
- Коррекция необходима, так как результат может превышать число 9.
- Если при сложении тетрад в входного переноса в тетраду сумма лежит в пределах от 10 до 15, необходимо организовать перенос в старшую тетраду с одновременным уменьшением полученной суммы на 10.

# Двоично-десятичные сумматоры

---

- Вычитание можно заменить сложением с дополнительным кодом числа 10, это число имеет двоичное представление 0110. Таким образом, в данном случае к результату нужно прибавить корректирующую поправку 6.

# Двоично-десятичные сумматоры





- 
- Сумматор с индексом 1 формирует значения аргументов, которые могут лежать в диапазоне от 0 до 31. Для обнаружения сумм, превышающих число 9 введены два логических элемента «И», которые в этих комбинациях выдают логическую единицу, поступающую на логический элемент «ИЛИ».

- 
- Выход элемента «ИЛИ» формирует выходной перенос в следующую тетраду.
  - Этот же сигнал логической единицы порождает добавление кода 0110 (по входам и ). При результате сложения на первом сумматоре больше числа 15 выдаётся выходной перенос, который поступает на следующую тетраду.

# Сумматоры накапливающего типа

---

- В архитектурах вычислительных устройств встречаются сумматоры, не относящиеся к сумматорам комбинационного типа. Одними из них являются накапливающие сумматоры.

- 
- Под термином «накапливающие» встречаются сумматоры двух разновидностей:
  - - сумматоры, построенные на основе счётного триггера и логических элементов,

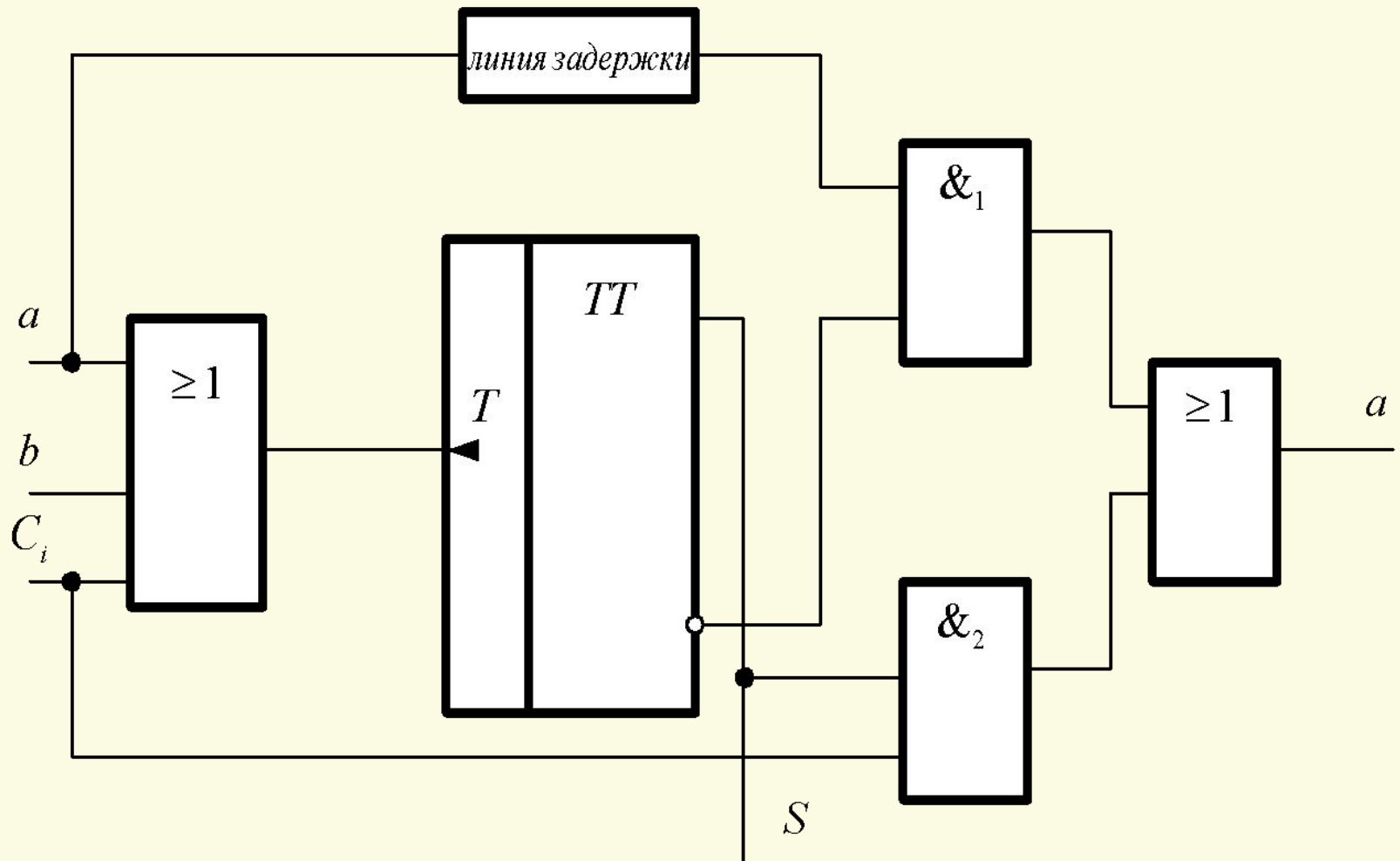
- 
- - сумматор со структурой «комбинационный многоразрядный сумматор плюс регистр хранения».

## Первая разновидность накапливающего сумматора на базе счётного Т-триггера

---

- Сумматор первого типа является медленным вследствие двухтактности операции сложения и сложностью управления и обнуления.

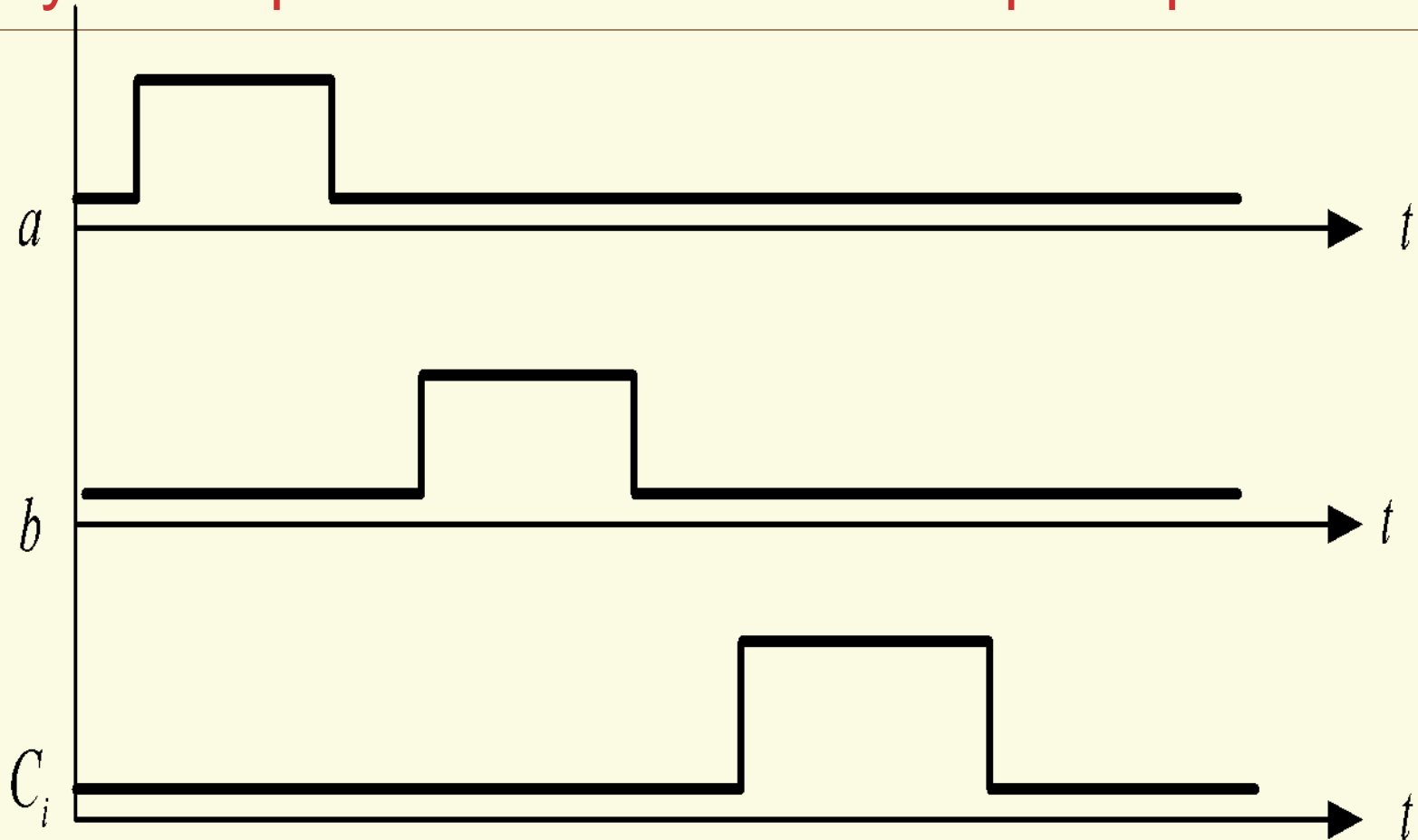
# Первая разновидность накапливающего сумматора на базе счётного Т-триггера



- 
- Особенностью является необходимость последовательной подачи во времени аргументов и входного переноса. Это требование вытекает из конечного времени переключения триггера. Последовательность подачи входных сигналов иллюстрирует временная диаграмма



# Первая разновидность накапливающего сумматора на базе счётного Т-триггера



- 
- В основе суммирования лежит свойство счётного триггера, которые меняет своё состояние на противоположное при каждой поступающей логической единицей на его счётный вход.

- 
- Двоичный одноразрядный код аргументов и входящего переноса поступают на счётный вход триггера последовательно согласно временной диаграмме. Значение суммы снимается с прямого выхода триггера после поступления перечисленных трёх сигналов.

- 
- Логические элементы «И» с номерами 1 и 2 формируют значения переноса, возникающее из-за сложения двух единичных аргументов или при сложении одного из единичного аргумента и единичного значения входного переноса.

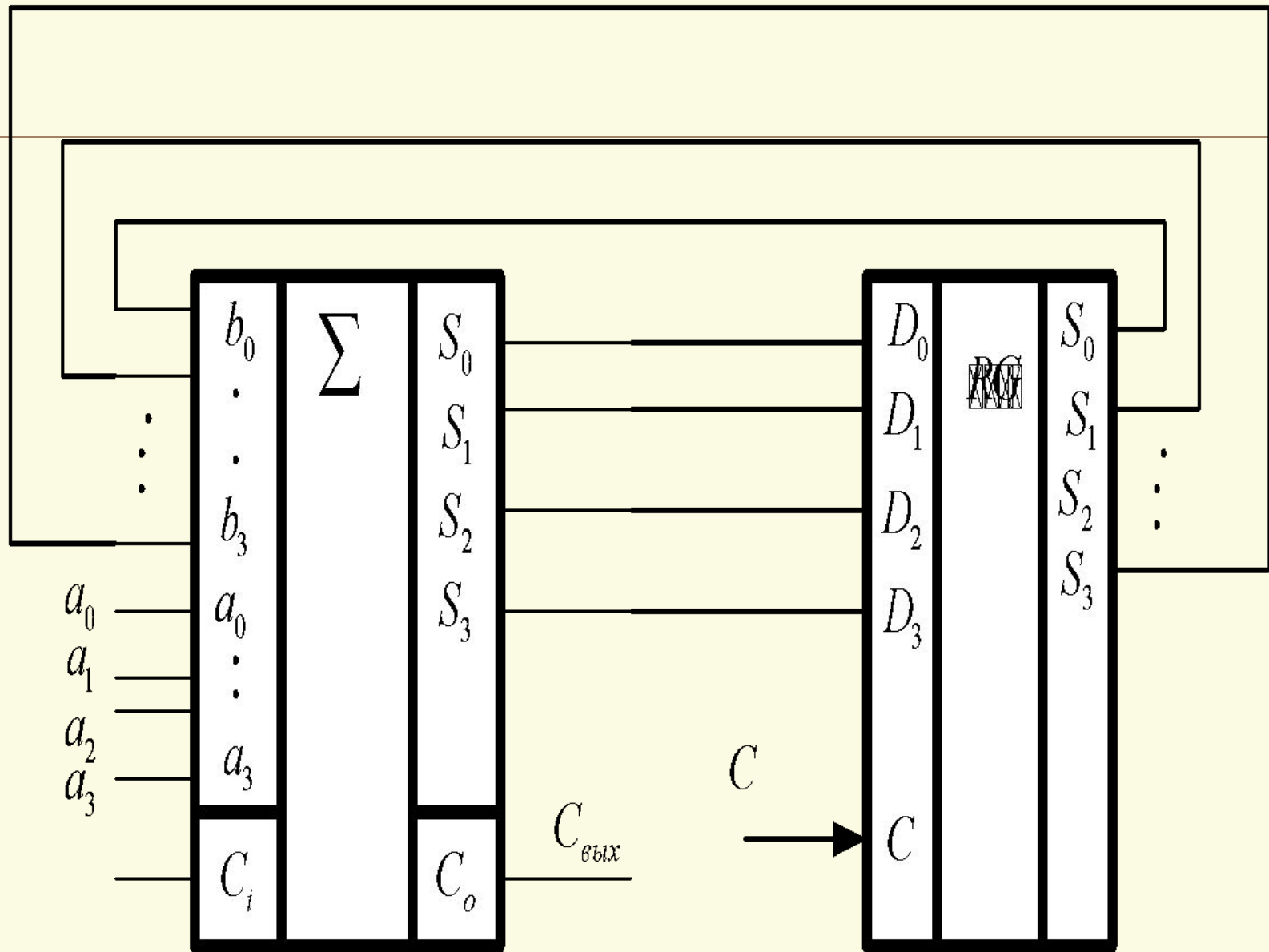
- 
- Линия задержки нужна для того, чтобы передать предыдущее состояние триггера в момент поступления второго аргумента для порождения выходного переноса. Логический элемент «И» с номером 2 порождает входной перенос при наличии входного переноса и одного единичного операнда.

## Вторая разновидность накапливающего сумматора

---

- Вторая разновидность накапливающего сумматора рис. 9.14. состоит из многоразрядного комбинационного сумматора, выходы которого соединяются со входами регистра хранения. Регистр хранения коммутируется своими выходами на вход второго аргумента сумматора.

- 
- Аргумент А подаётся на вход первого аргумента сумматора. При каждом тактовом импульсе регистр обновляет своё содержимое новой промежуточной суммой .





# Классификация и основные типы вычитателей.

---

- ***Вычитатель*** – устройство комбинационного типа, предназначенное для выполнения операции вычитания двоичных чисел.

# Полувычитатель.

---

- **Полувычитатель - это комбинационная схема, которая вырабатывает сигналы разности и заем при вычитании двух двоичных чисел .**

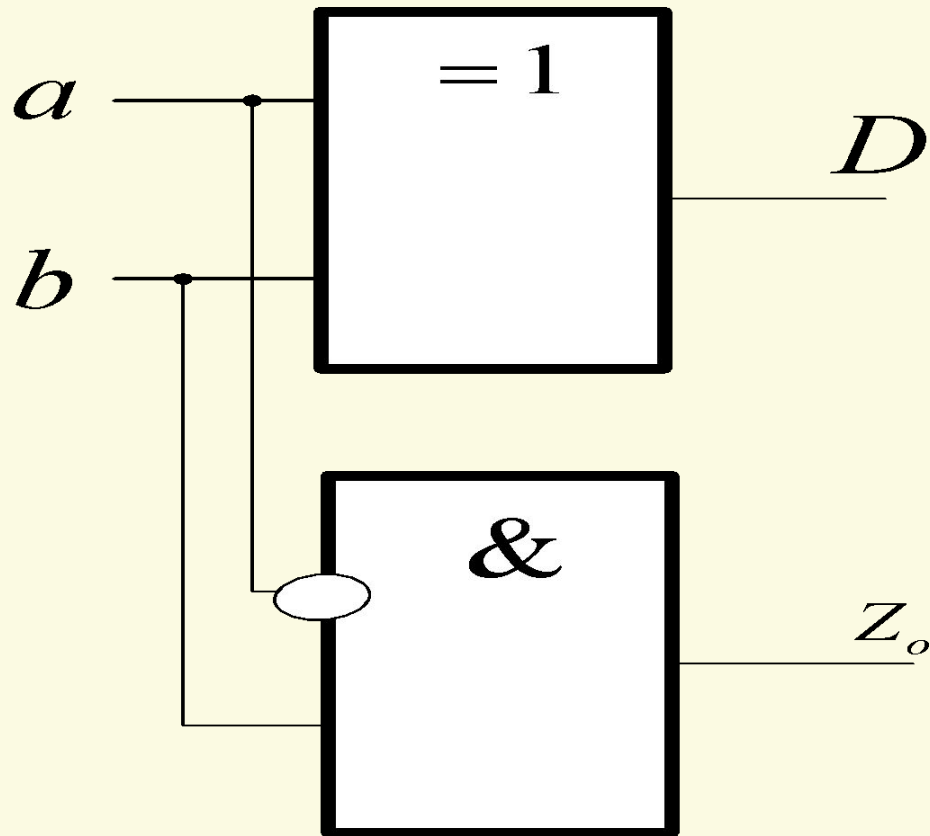
# Таблица одnorазрядного вычитателя

---

a	b	D (разность)	Z (заём)
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

- 
- Из таблицы следует, что функция разности реализуется логическим элементом «Исключающее ИЛИ», а функция заёма реализуется логическим элементом «Запрет».

# Схема полувывчитателя

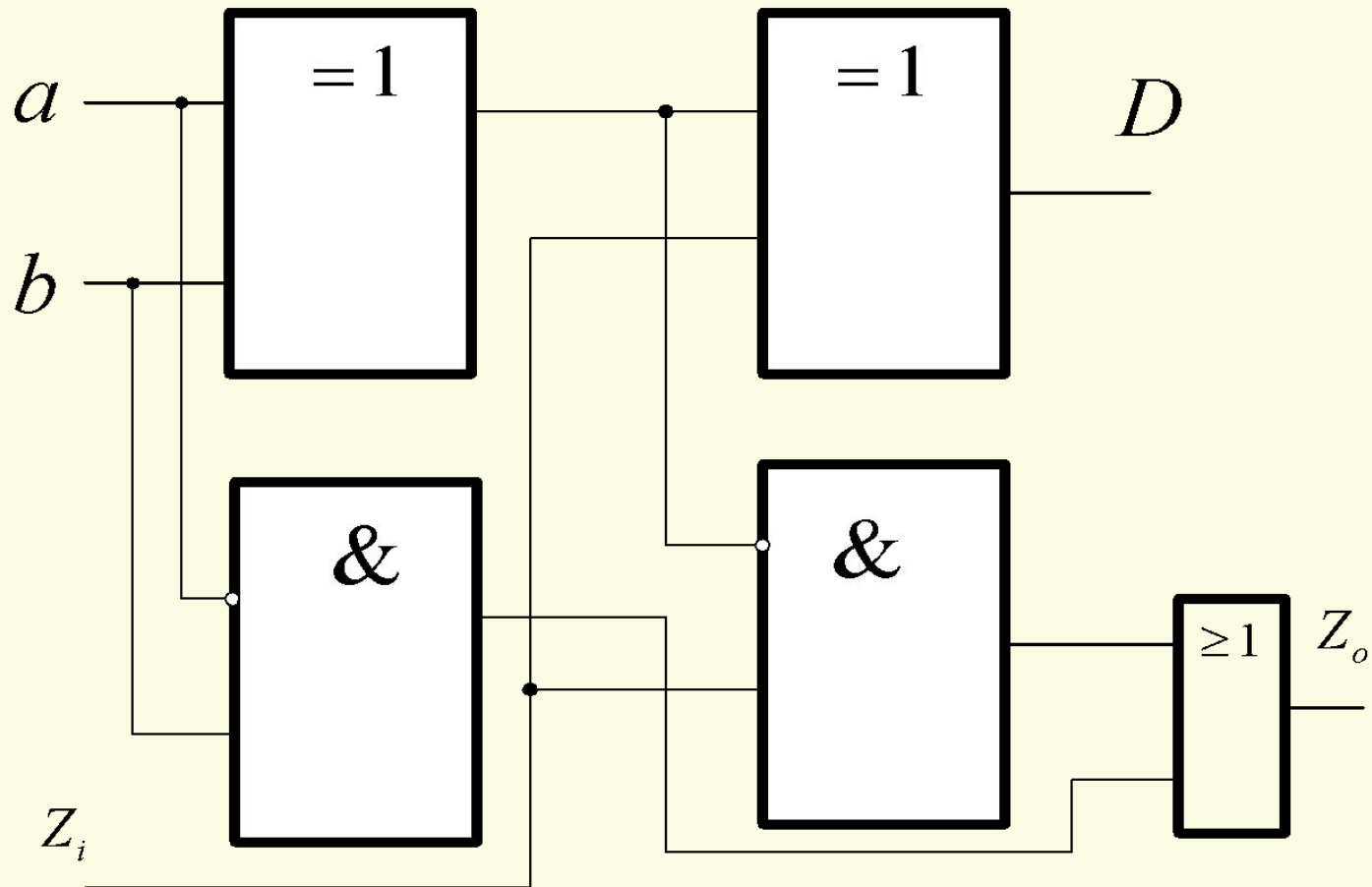


# Полный вычитатель.

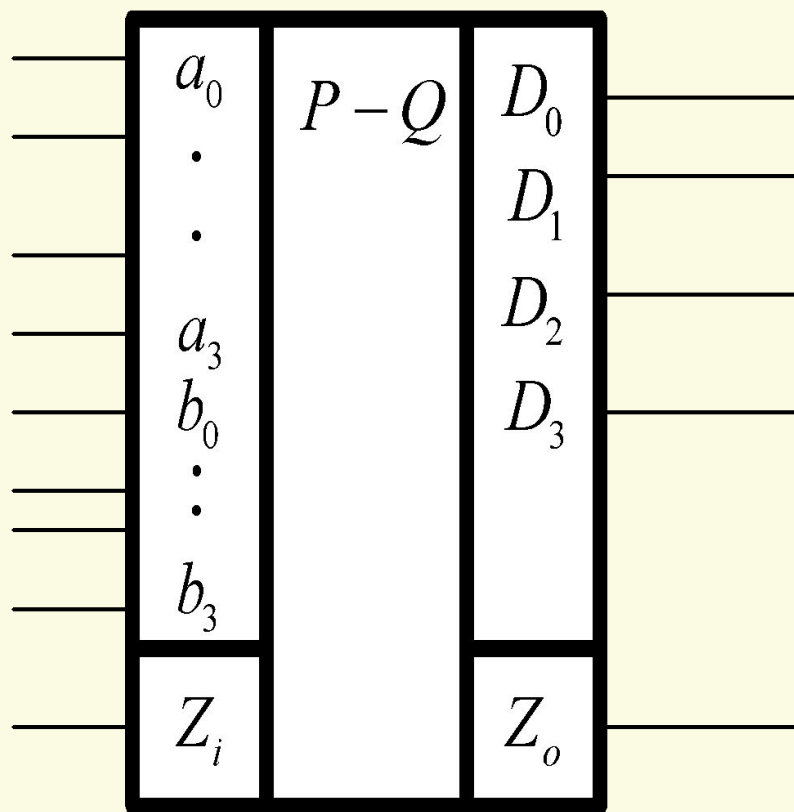
---

- Полный вычитатель имеет помимо входов аргументов и третий вход входного заёма из младшего разряда .

# схема полного вычитателя на элементах «Исключающее ИЛИ» и «Запрет»

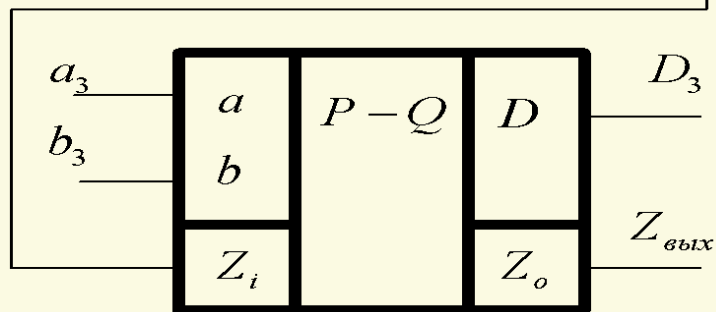
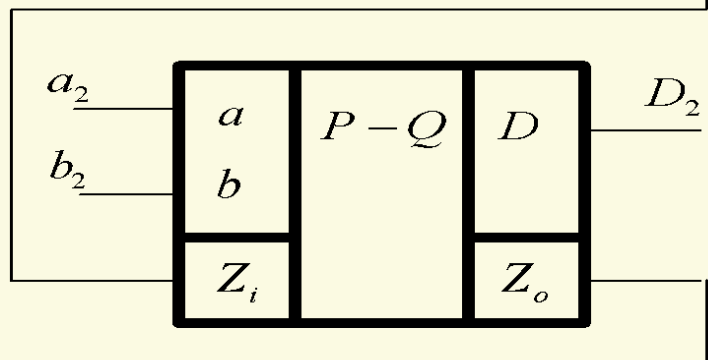
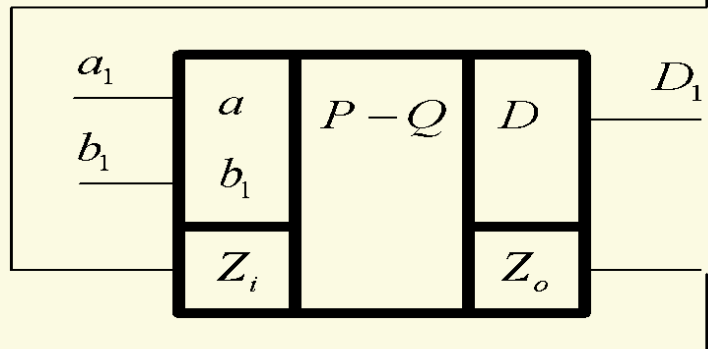
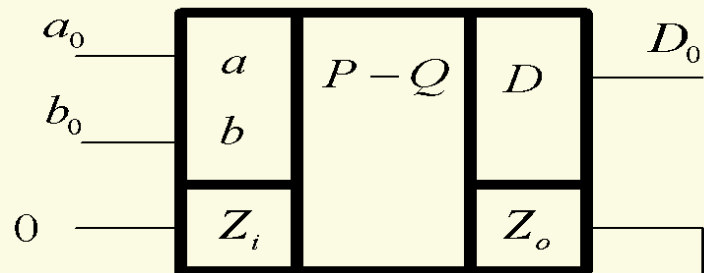


# Условно-графическое изображение вычитателя





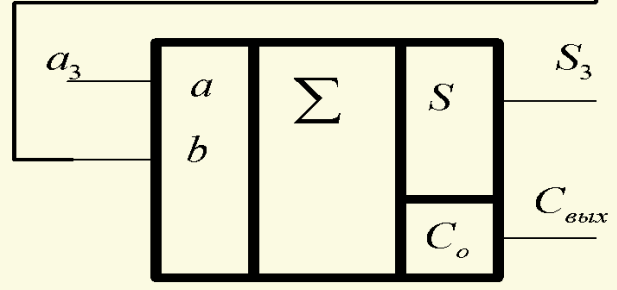
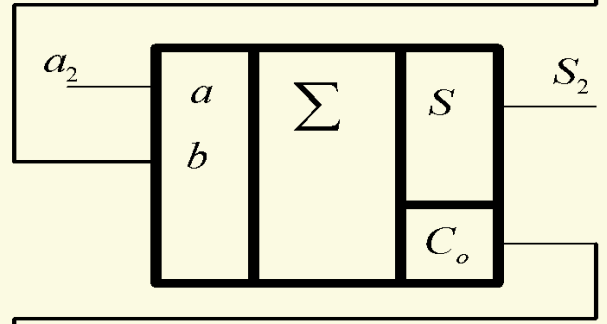
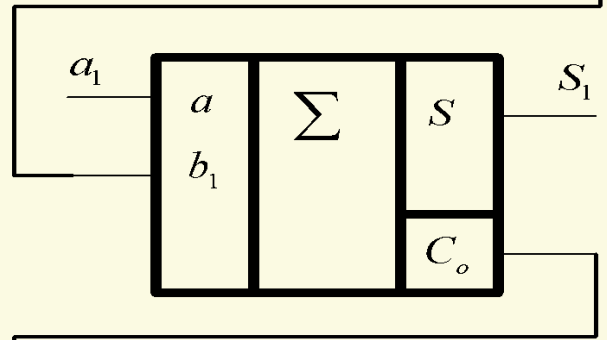
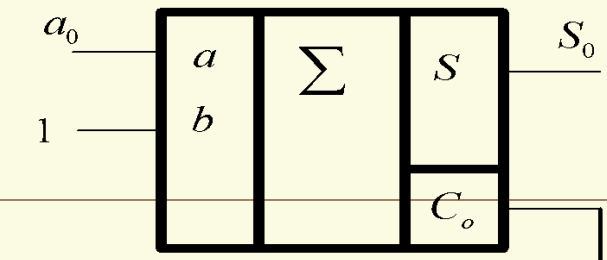
- 
- Аналогично сумматору параллельного действия с последовательным переносом можно построить схему параллельного многоразрядного вычитателя с последовательным заёмом



# Инкременторы и декременторы

---

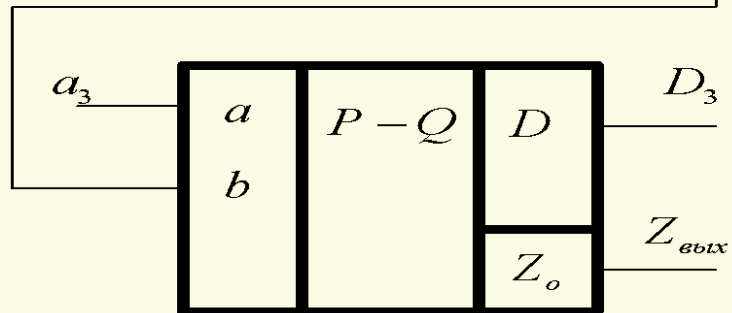
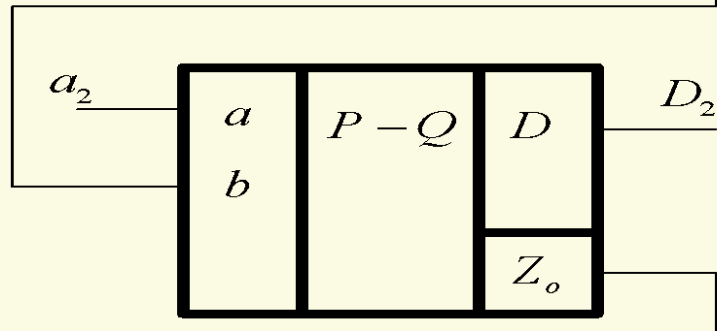
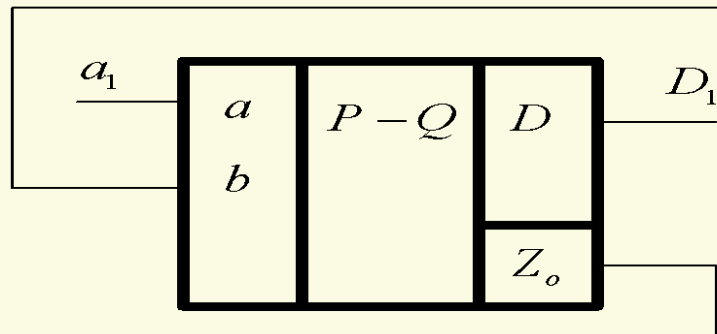
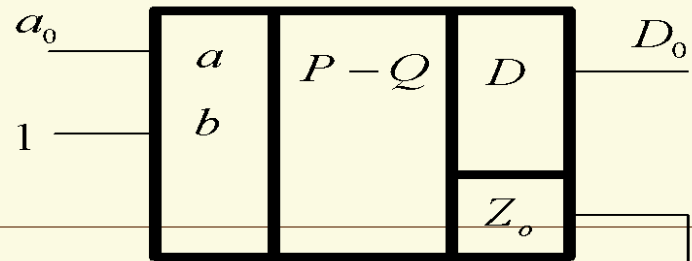
- ***Инкрементор*** - это комбинационная схема осуществляющая сложение многоразрядного двоичного кода аргумента  $A$  с значением одноразрядной логической единицы.



# Инкременторы и декременторы

---

- ***Декрементор*** - это комбинационная схема осуществляющая вычитание из многоразрядного двоичного кода аргумента  $A$  значения одноразрядной логической единицы.



## Сумматоры дополнительного кода и сумматоры обратного кода.

---

- В большинстве вычислительных устройств используется представление целых чисел со знаком. Старший разряд используется для кодирования знака: 0 обозначает положительное число, 1- отрицательное число.

- 
- . Использование представления чисел вместе со знаками в двоичном коде позволяет осуществлять арифметические операции суммирования и вычитания, получая результат с готовым знаком числа результата, применяя тот же классический комбинационный многоразрядный сумматор.



- 
- Применение дополнительного и обратного кода позволяет представлять отрицательные числа в этих кодах и, подавая их на сумматор фактически выполняя операцию суммирования, получать разность.

- 
- Перед рассмотрением примеров использования дополнительного и обратного кода необходимо остановится на следующих аксиомах и правилах:

- 
- Сумма двух чисел в дополнительном (или обратном) коде есть дополнительный (или обратный) код.
  - Дополнительный (обратный) код положительного числа совпадает с его прямым кодом.

- 
- Дополнительный код отрицательного числа со знаком формируется путём инвертирования разрядов числовой части целого числа, исключая знак и арифметическое сложение инвертированного значения с единицей.

- 
- Обратный код отрицательного числа со знаком формируется путём инвертирования разрядов числа, кроме разряда знака.

- 
- Сумматор дополнительного кода выглядит как многоразрядный комбинационный сумматор

- 
- Самый старший одноразрядный сумматор в таком сумматоре складывает двоичные коды знаков. Выходной перенос из знакового разряда отбрасывается.

## примеров сложения чисел со знаками в различных сочетаниях у аргументов.

- 9.6.1.  $C=A+B=(+9)+(-3)$

- В прямом коде: A 0 1001 ; B 1 0011 .

- В дополнительном коде A 0 1001 ; B 1 1101 .

- Результат сложения: 0 1001

+

1 1101

- C= 0 0110 наличие нуля в знаковом разряде говорит о положительном результате. То есть дополнительный код результата равен прямому коду и это есть окончательный результат.



● 9.6.2  $C=A+B=(+4)+(-9)$

● В прямом коде: A 0 0100 ; B 1 1001

● В дополнительном коде: A 0 0100 ; B  
0111

● Результат сложения: 0 0100

● +  
1 0111

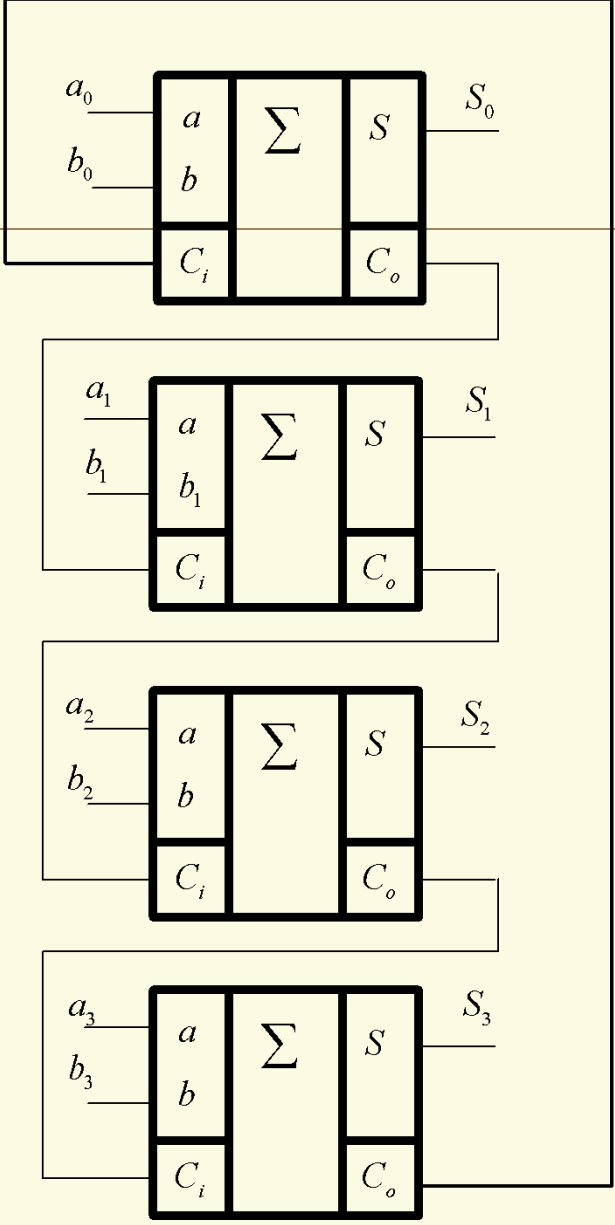
●  $C=1\ 1011$  (отрицательный  
результат в дополнительном коде).

- 
- После преобразования в прямой код  $C = 1\ 0101$ .
  - Операция сложения со знаками возможна и в обратных кодах.

# Сумматор обратного кода

---

- Особенностью сумматора обратного кода является наличие связи по переносу между сумматором знаков и сумматором младших разрядов.  
Схема сумматора обратного кода



# Применение модифицированных кодов.

---

- При выполнении операций в дополнительном и обратном коде включая знаки чисел происходит нарушение правильного значения знака результата из-за переполнения разрядной сетки числовой части и поступления вызванного переполнением переноса в сумматор знаков.

- 
- Аппаратно определить данную ситуацию не представляется возможным. Для решения данной проблемы было предложено использовать модифицированный дополнительный и модифицированный обратный код.

- 
- Отличие состоит в дублировании битов, обозначающих код знаков слагаемых. Благодаря такому дублированию, возникающий перенос искажает только один из знаков результата в модифицированном коде.

- 
- Признаком переполнения является несовпадение знаков результата выполнения операции сложения. Это несовпадение легко обнаруживается при помощи логического элемента «Исключающее ИЛИ».



# Пример выполнения операций с модифицированными кодами

---

- $A = +9$  и  $B = +8$
- 00 1001
- +
- 00 1000
- = 01 0001 при сложении двух прямых кодов получились разные биты знаков, что свидетельствует о переполнении

---

- $A = +4 \quad B = -7$

- $A_{\text{обр}} = 00\ 0100 \quad B_{\text{обр}} = 11\ 1000$

- $00\ 0100$

- $+$

- $11\ 1000$

- $11\ 1100$  модиф обр код

- $11\ 0011$  модиф. прямой код

---

● A=-10    B=-8

● A    11 1010      B    11 1000

●      11 0101

●    +

●      11 0111

●      10 1101

# Построение инкремента

---

- ***Многоразрядный инкрементор*** строится из  $n$  полусумматоров, объединяемых трактом последовательного переноса.
- При этом вход переноса осуществляется на вход второго слагаемого в полусумматор.
- На один из входов младшего полусумматора инкрементора подаётся логическая единица.

# Построение декрементора

---

- ***Многоразрядный декрементор*** строится из  $n$  полувывчитателей, объединяемых трактом последовательного заёма.
- При этом вход заёма осуществляется на вход вычитаемого в полувывчитатель.
- На один из входов младшего полувывчитателя декрементора подаётся логическая единица.

# Применение инкременторов и декременторов

---

- Инкременторы и декременторы используются, например при организации серий обращений к последовательным адресам памяти.
- Для этого они вводятся в состав микропроцессоров. Они выполняют функции счётчиков, но без запоминания результата и соответственно без потери исходного числа.
- Оба числа:  $A$  и  $A+1$  **существуют одновременно**, одно на входе, другое на выходе схемы.

# Компараторы

---

- Компараторы относятся к арифметическим устройствам.
- Они **выполняют сравнение двух чисел**, заданных в двоичных (либо других) кодах.
- Компаратор выдаёт сигналы на своих выходах о равенстве равноразрядных чисел  $A$  и  $B$ , либо их неравенство ( $A < B$ ,  $A > B$ )

# Компараторы

---

- ***Компараторы применяются:***
- - для выявления нужного числа в потоке чисел,
- - для отметки времени в часовых приборах,
- - для выполнения условного перехода в вычислительных устройствах.



# Компараторы

---

- В *устройствах автоматики* компараторы применяются:
- - для сигнализации о выходе величины за пределы допуска;
- - в приводах следящих систем для определения направления воздействия ликвидирующего рассогласование;
- - при построении счётчиков по произвольному основанию.

# Синтез одnorазрядного компаратора

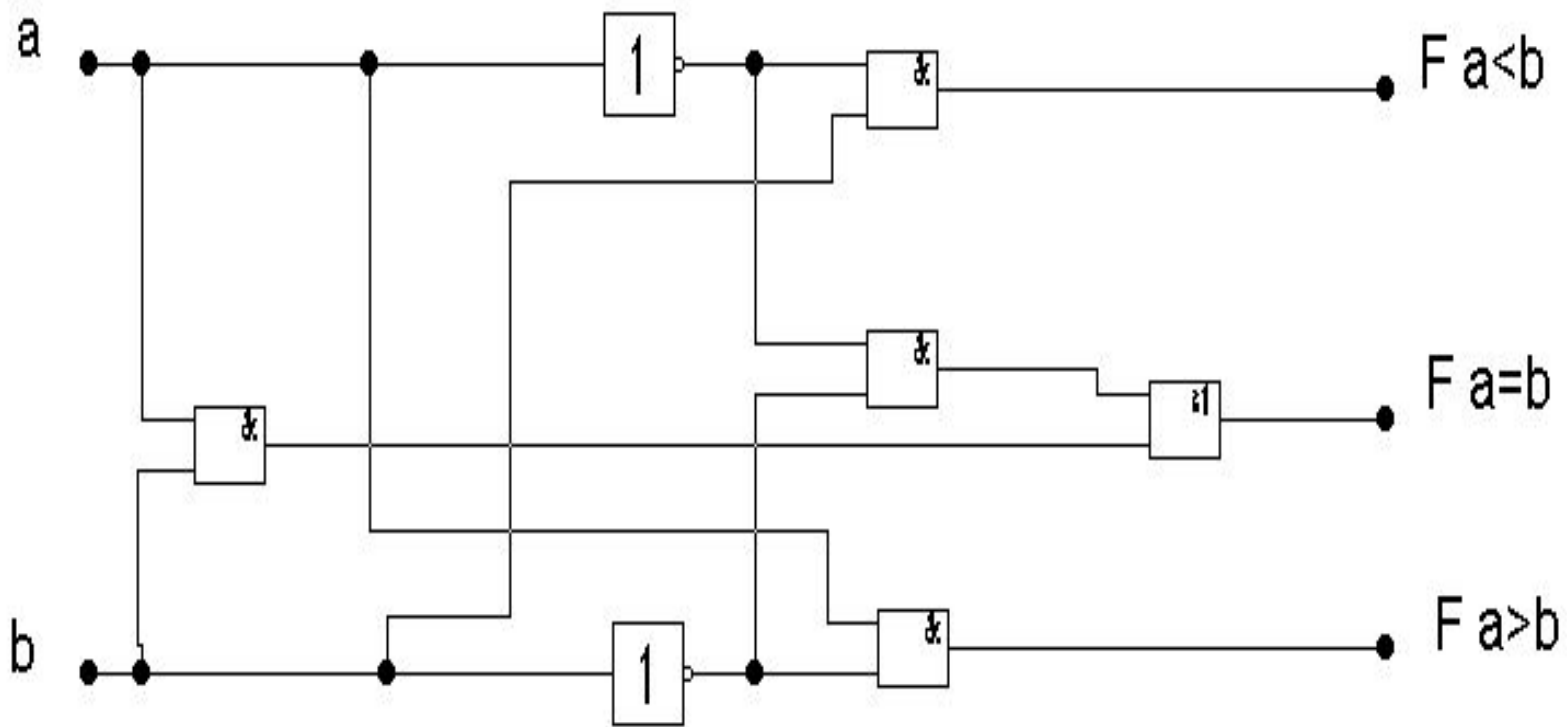
a	b	F a>b	F a=b	F a<b
1	1	0	1	0
1	0	1	0	0
0	1	0	0	1
0	0	0	1	0

# Синтез одноразрядного компаратора

---

- На соответствующем выходе компаратора 1, когда сравниваемые входные двухразрядные числа находятся в нужном соотношении.
- $F_{a>b} = a!b$ ,  $F_{a=b} = ab + !a!b$ ,
- $F_{a<b} = !ab$

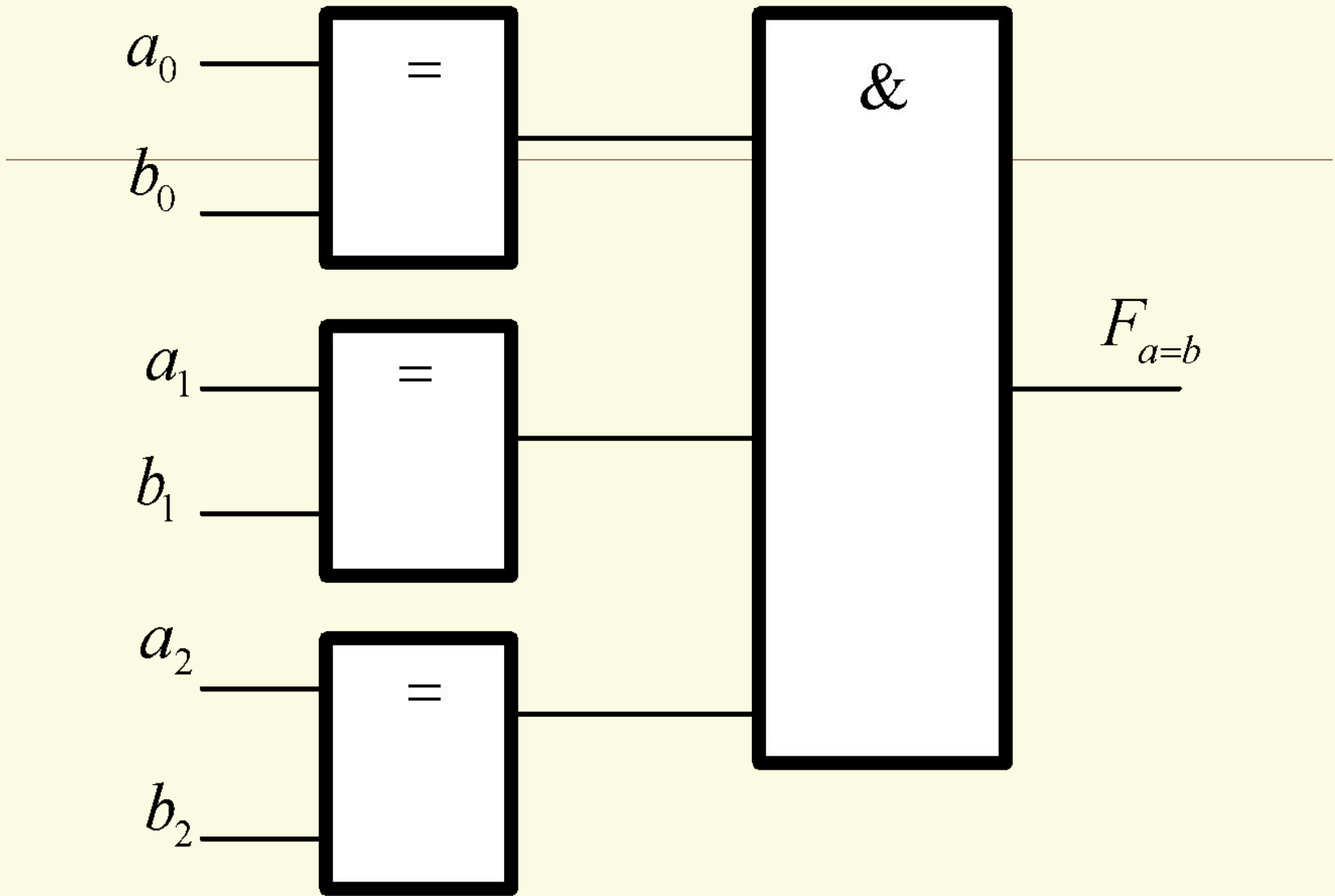
# Схема одноразрядного компаратора



# Многоразрядные компараторы

---

- Трёхразрядный компаратор, выполняющий сравнение двух трёхразрядных кодов на равенство



- 
- При сравнении разряда на выходе логического элемента «равнозначность» выдаётся логическая единица. Выходы всех элементов «равнозначность» сведены на входы логического элемента «И». На выходе логического элемента «И» логическая единица выдаётся при равенстве всех разрядов.

# Синтез многоразрядного компаратора

---

- Основывается на анализе влияния анализируемого разряда на выдаваемого функцию сравнения. Рассмотрим пример синтеза компаратора на три разряда.



# Синтез многоразрядного компаратора

---

- Часть схемы компаратора, выдающая сигнал равенства двух трёхразрядных чисел соответствует схеме сравнения на равенство. Если осуществить запись выражений для одного из неравенств, то для определения второго сигнала неравенства можно принять, что это будет при нулевом значении противоположного неравенства и равенства.

# Синтез многоразрядного компаратора

---

- Если выходы этих функций подать на вход логического элемента «ИЛИ-НЕ» получим единичное значение искомой функции неравенства.

# Синтез многоразрядного компаратора

- Разобьем синтез схемы для получения, например, функции на три этапа анализа влияния каждого из трёх разрядов:
- - этап 1. Сравниваются наиболее значимые биты, т.е.  $a_2$  с  $b_2$ . Если ,  $a_2 > b_2$
- то и  $A > B$  Следовательно, схема для этого случая синтезируется на основе выражения 
$$F_{2_{a>b}} = a_2 \overline{b_2} \quad ;$$

# Синтез многоразрядного компаратора

---

- - этап 2. Если старшие разряды равны  $a_2 = b_2$ , то  $F_{a>b}$  выполнится при  $a_1 > b_1$ . Следовательно, схема для этого случая синтезируется на основе выражения

- $$F1_{a>b} = (a_2 b_2 + \bar{a}_2 \bar{b}_2) * a_1 \bar{b}_1 ;$$

# Синтез многоразрядного компаратора

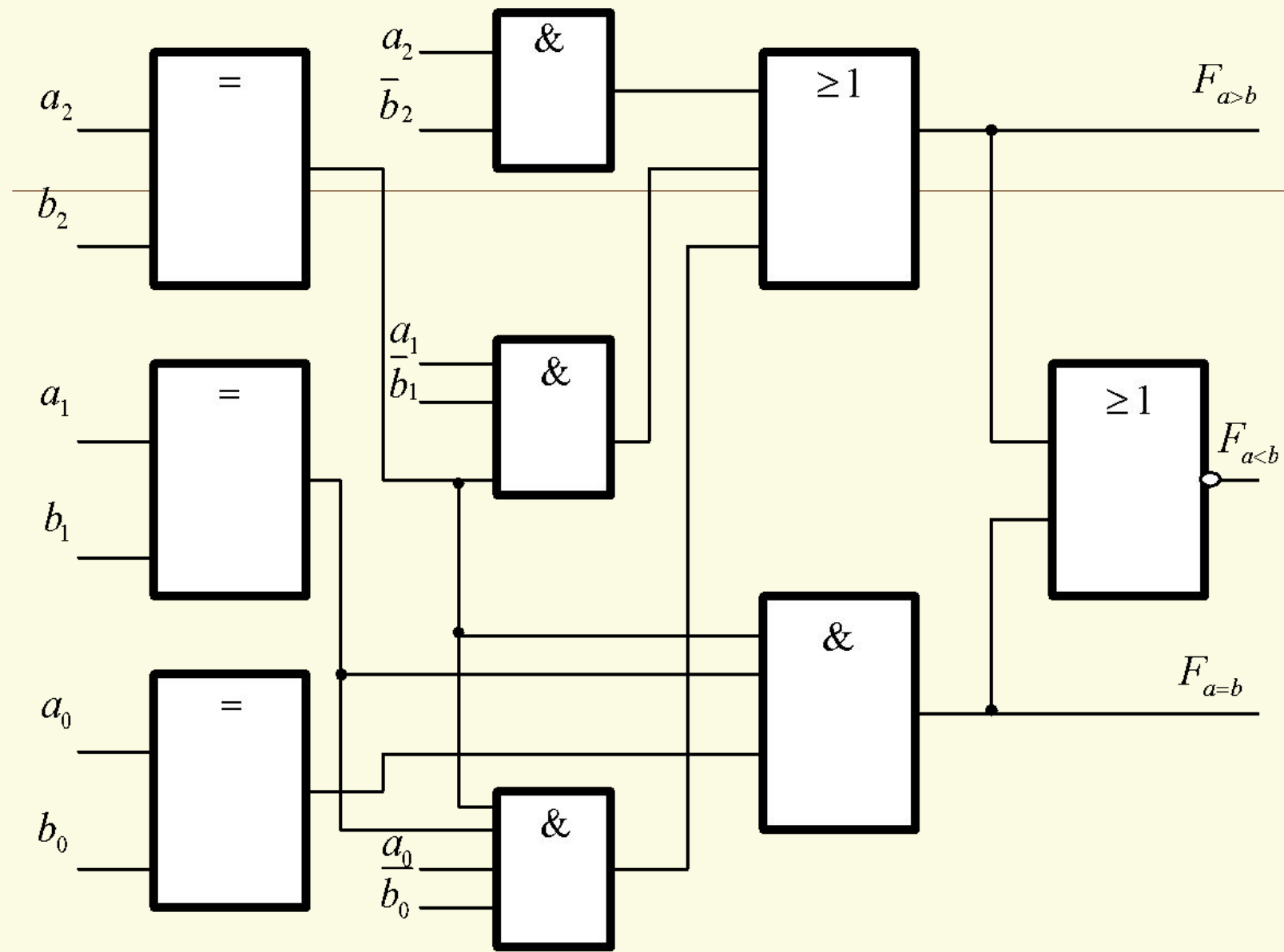
---

- - этап 3. Если оба разряда  $a_2 = b_2$  и
- $a_1 = b_1$ , то  $F_{a>b}$  выполнится при
- $a_0 > b_0$ . Следовательно, схема для этого случая синтезируется на основе выражения
- $$F_{a>b} = F2_{a>b} + F1_{a>b} + F0_{a>b} \quad .$$

# Синтез многоразрядного компаратора

---

- Все эти три выражения по трём этапам связываются функцией дизъюнкцией:
- $$F_{a>b} = F2_{a>b} + F1_{a>b} + F0_{a>b}$$
- Согласно выведенным выражениям схема трёхразрядного компаратора для трёх функций условий сравнения примет вид на очередном слайде:



# Завершение темы «арифметические устройства»

---

- Всем спасибо за внимание.