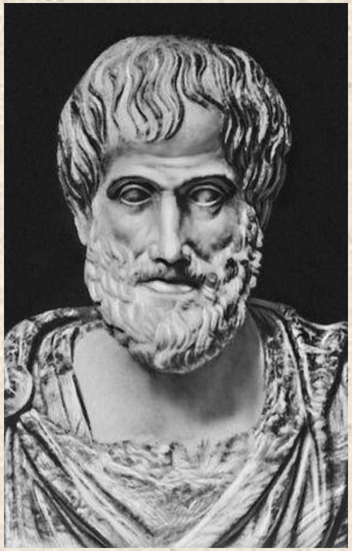
АЛГЕБРА ЛОГИКИ

МАТЕМАТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ

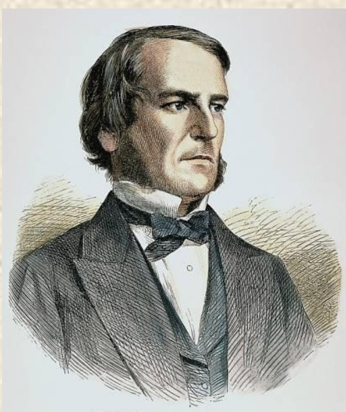
Логика – это наука о формах и законах человеческого мышления.



Ее основоположник – древнегреческий мыслитель
Аристотель (384-322 года до н. э.).

Логика

Вильгельм Лейбниц (1646-1716). Основоположник математической логики (пытался построить первые логические исчисления: арифметические и буквенно-алгебраические).



Джордж Буль (1815-1864). Создал новую область науки - Алгебру логики (Булеву алгебру или Алгебру высказываний).

Алгебра логики (булева алгебра) - это раздел математики, изучающий высказывания, и логические операции над ними.

Цель алгебры логики – описание поведения и структуры логических схем.

Объекты алгебры логики – **высказывания**.

Логическое высказывание – это повествовательное предложение, относительно которого можно однозначно сказать, истинно оно или ложно.

Высказывание или нет?

- Сейчас идет дождь.
- Жирафы летят на север.
- История – интересный предмет.
- У квадрата – 10 сторон и все разные.
- Красиво!
- В городе N живут 2 миллиона человек.
- Который час?

Виды высказываний

Высказывания

Простые

Составные

Высказывание называется **простым**, если никакая его часть сама не является высказыванием.

Сложные (составные) высказывания строятся из простых с помощью логических связок (и; или; не; если, то; и др).

Так, например, из элементарных высказываний "Петров — врач", "Петров — шахматист"
при помощи связки "и" можно получить составное высказывание
"Петров — врач и шахматист", понимаемое как "Петров — врач, хорошо играющий в шахматы".

При помощи связки "или" из этих же высказываний можно получить составное высказывание
"Петров — врач или шахматист",
понимаемое в алгебре логики как
"Петров или врач, или шахматист, или и врач и шахматист одновременно".

В алгебре логики высказывания обозначают ЗАГЛАВНЫМИ буквами латинского алфавита и называют **логическими переменными**.

Если высказывание истинно, то значение соответствующей ему логической переменной обозначают единицей (**А** = **1**), а если ложно - нулём (**В** = **0**).

0 и **1** называются **логическими значениями**.

Так, например, предложение
" Трава зеленая" следует считать высказыванием, так как оно истинное.
Записывается: А=1

Предложение " Лев - птица" тоже высказывание, так как оно ложное.
Записывается: В=0

Пусть через A обозначено высказывание "Тимур поедет летом на море", а через B — высказывание "Тимур летом отправится в горы".

Тогда составное высказывание "Тимур летом побывает и на море, и в горах" можно кратко записать как

А и В

Здесь "и" — логическая связка, А, В — логические переменные, которые могут принимать только два значения - "истина" или "ложь", обозначаемые, соответственно, "1" и "0".

Составьте из простых высказываний составные при помощи связок:

А – Сейчас идет дождь.

В – Форточка открыта.

А и В	Сейчас идет дождь и открыта форточка.
А или не В	Сейчас идет дождь или форточка закрыта.
если А, то В	Если сейчас идет дождь, то форточка открыта.
не А и В	Сейчас нет дождя и форточка открыта.
А тогда и только тогда, когда В	Дождь идет тогда и только тогда, когда открыта форточка.

Операция НЕ

Операция, выражаемая словом "не", называется **инверсией** или отрицанием и обозначается чертой над высказыванием.

$\overline{A}$

Если высказывание **А** истинно,

то "**не А**" ложно, и наоборот.

Высказывание $\overline{A}$ истинно, когда A ложно, и ложно, когда A истинно.

Пример. "Луна — спутник Земли" (A);

"Луна — не спутник Земли" ($\overline{A}$).

Таблица истинности

логического выражения F – это таблица, где в левой части записываются все возможные комбинации значений исходных данных, а в правой – значение выражения F для каждой комбинации.

A	не A
0	1
1	0

таблица истинности операции НЕ

Операция И

Операция, выражаемая связкой "и", называется **<u>конъюнкцией</u>**

(лат. conjunctio — соединение)

или ***логическим умножением***

и обозначается точкой " · "

(может также обозначаться знаками ^ или &).

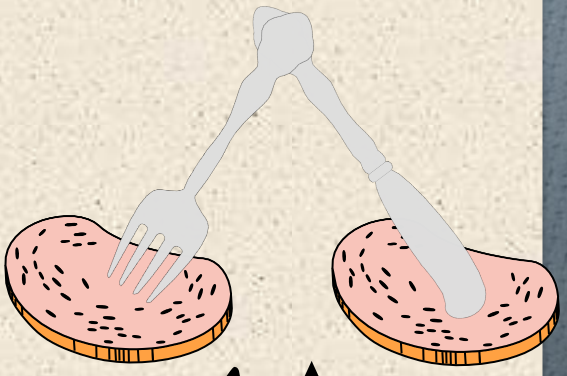
Высказывание $A \cdot B$ истинно тогда и только тогда, когда оба высказывания A и B истинны.

Например, высказывание
"10 делится на 2 и 5 больше 3" истинно,
а высказывания:
"10 делится на 2 и 5 не больше 3",
"10 не делится на 2 и 5 больше 3",
"10 не делится на 2 и 5 не больше 3"
— ложны.

Таблица истинности конъюнкции

A	B	A и B
0	0	0
0	1	0
1	0	0
1	1	1

также: **A·B**, **A ∧ B**, **A & B**



A ∧ B

Операция ИЛИ

Операция, выражаемая связкой "или" называется <u>дизъюнкцией</u>

(лат. disjunctio — разделение)

или *логическим сложением* и обозначается знаком **V** (или **+** «плюсом»).

Высказывание A v B <u>ложно</u> тогда и только тогда, когда <u>оба</u> высказывания A и B <u>ложны</u>.

Например, высказывание

"10 не делится на 2 или 5 не больше 3" ложно, а высказывания

"10 делится на 2 или 5 больше 3",

"10 делится на 2 или 5 не больше 3",

"10 не делится на 2 или 5 больше 3"— истинны.

Таблица истинности дизъюнкции

A	B	A или B
0	0	0
0	1	1
1	0	1
1	1	1

также: **A+B**, **A ∨ B**

Базовый набор операций

С помощью операций **И**, **ИЛИ** и **НЕ** можно реализовать любую логическую операцию.

И

ИЛИ

НЕ

базовый набор операций

? **Сколько всего существует логических операции с двумя переменными?**

Операция ЕСЛИ-ТО

Операция, выражаемая связками "если ..., то", "из ... следует", "... влечет ...",

называется **<u>импликацией</u>**

(лат. *implico* — тесно связаны) и обозначается знаком $\rightarrow$. Высказывание **A** $\rightarrow$ **B** ложно тогда и только тогда, когда **A** истинно, а **B** ложно.

Таблица истинности импликации

А – "Работник хорошо работает".

В – "У работника хорошая зарплата".

A	B	$A \rightarrow B$
0	0	1
0	1	1
1	0	0
1	1	1

Операция РАВНОСИЛЬНО

Операция, выражаемая связками "*тогда и только тогда*", "*необходимо и достаточно*", "*... равносильно ...*", называется **<u>эквиваленцией</u>** и обозначается знаком или ~. $\leftrightarrow$

Высказывание **A** $\leftrightarrow$ **B** истинно тогда и только тогда, когда значения **A** и **B** совпадают.

Таблица истинности эквиваленции

A	B	$A \leftrightarrow B$
0	0	1
0	1	0
1	0	0
1	1	1

С помощью логических переменных и символов логических операций любое высказывание можно формализовать, то есть заменить **логической формулой.**

Определение **логической формулы**:

1. Всякая логическая переменная и символы "истина" ("1") и "ложь" ("0") - формулы.

2. Если A и B - формулы, то $\overline{A}$, $A \cdot B$, $A \vee B$, $A \rightarrow B$, $A \leftrightarrow B$ - формулы.

3. Никаких других формул в алгебре логики нет.

Порядок выполнения логических операций в сложном логическом выражении

1. Инверсия;
2. Конъюнкция;
3. Дизъюнкция;
4. Импликация;
5. Эквивалентность.

Определите истинность составного высказывания:

$(\overline{A}\&\overline{B})\ \&\ (C \vee D)$, состоящего из простых высказываний:

A = {Принтер – устройство вывода информации},
B = {Процессор – устройство хранения информации},
C = {Монитор – устройство вывода информации},
D = {Клавиатура – устройство обработки информации}.

Сначала на основании знания устройства компьютера устанавливаем истинность простых высказываний:

A = 1, *B* = 0, *C* = 1, *D* = 0.

Определим теперь истинность составного высказывания, используя таблицы истинности логических операций:

$(\overline{1}\ \&\ \overline{0})\ \&(1 \vee 0) = (0\ \&\ 1)\ \&\ (1 \vee 0) = 0$

Составное высказывание ложно.

Даны простые высказывания:

A = {Принтер – устройство ввода информации},
B = {Процессор – устройство обработки информации},
C = {Монитор – устройство хранения информации},
D = {Клавиатура – устройство ввода информации}.

Определите истинность составных высказываний:

а) $(A \& B) \& (C \vee D)$;

б) $(A \& B) \Rightarrow (C \vee D)$;

в) $(A \vee B) \Leftrightarrow (C \& D)$;

г) $\overline{A} \Leftrightarrow \overline{B}$.

Логические выражения и их таблицы истинности

Составные высказывания в алгебре логики записываются с помощью логических выражений. Для любого логического выражения достаточно просто построить таблицу истинности.

$$F = A \cdot B + A \cdot C + B \cdot C$$

логическая формула

Таблица истинности - это табличное представление логической схемы (операции), в котором перечислены все возможные сочетания значений истинности входных сигналов (операндов) вместе со значением истинности выходного сигнала (результата операции) для каждого из этих сочетаний.

Построение таблиц истинности для логических выражений

- подсчитать ***n*** - число переменных в выражении
- подсчитать общее число логических операций в выражении
- установить последовательность выполнения логических операций
- определить число столбцов в таблице
- заполнить шапку таблицы, включив в неё переменные и операции
- определить число строк в таблице без шапки: $m = 2^n$
- выписать наборы входных переменных
- провести заполнение таблицы по столбцам, выполняя логические операции в соответствии с установленной последовательностью

Пример построения таблицы истинности

A ∨ ***A*** & ***B***

n (число переменных) = 2,

m (количество строк без шапки)= 2^2 = 4.

Операций – 2, значит количество столбцов будет: *n+2=4*

Приоритет операций: &, V

A	*B*	*A&B*	*A*∨*A&B*
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

Составление таблиц истинности

$$F = A \cdot B + \overline{A} \cdot B + \overline{B}$$

	A	B	$A \cdot B$	$\overline{A} \cdot B$	$\overline{B}$	F
0	0	0	0	0	1	1
1	0	1	0	1	0	1
2	1	0	0	0	1	1
3	1	1	1	0	0	1

Составление таблиц истинности

$$F = A \cdot B + A \cdot C + B \cdot C$$

	A	B	C	A·B	A·C	B·C	F
0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0
2	0	1	0	0	0	0	0
3	0	1	1	0	0	1	1
4	1	0	0	0	0	0	0
5	1	0	1	0	1	0	1
6	1	1	0	1	0	0	1
7	1	1	1	1	1	1	1

ЗАКОНЫ АЛГЕБРЫ ЛОГИКИ

Закон тождества

A = A

Всякое понятие и суждение тождественно самому себе.

Закон тождества означает, что в процессе рассуждения нельзя подменять одну мысль другой, одно понятие другим. При нарушении этого закона возможны логические ошибки. Например, рассуждение *Правильно говорят, что язык до Киева доведет, а я купил вчера копченый язык, значит, теперь смело могу идти в Киев* неверно, так как первое и второе слова «язык» обозначают разные понятия.

Закон непротиворечия

A & notA = 0

Не могут быть одновременно истинны утверждение и его отрицание.

То есть если высказывание *А*— истинно, то его отрицание *не А* должно быть ложным (и наоборот). Тогда их произведение будет всегда ложным.
Это равенство часто используется при упрощении сложных логических выражений.
Примеры невыполнения закона непротиворечия:
1. *На Марсе есть жизнь и на Марсе жизни нет.*
2. *Оля окончила среднюю школу и учится в X классе.*

Закон исключения третьего

A and not A = 1

В один и тот же момент времени высказывание может быть либо истинным, либо ложным, третьего не дано.

Истинно либо *А*, либо не *А*.
Примеры выполнения закона исключенного третьего:

1. *Число 12345 либо четное, либо нечетное, третьего не дано.*
2. *Предприятие работает убыточно или безубыточно.*
3. *Эта жидкость является или не является кислотой.*

Закон двойного отрицания

Not (notA)=1

Если отрицать дважды некоторое высказывание, то в результате получается исходное высказывание.

Например, высказывание
А = Матроскин — кот эквивалентно высказыванию

А = Неверно, что Матроскин не кот.

ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ КОМПЬЮТЕРА

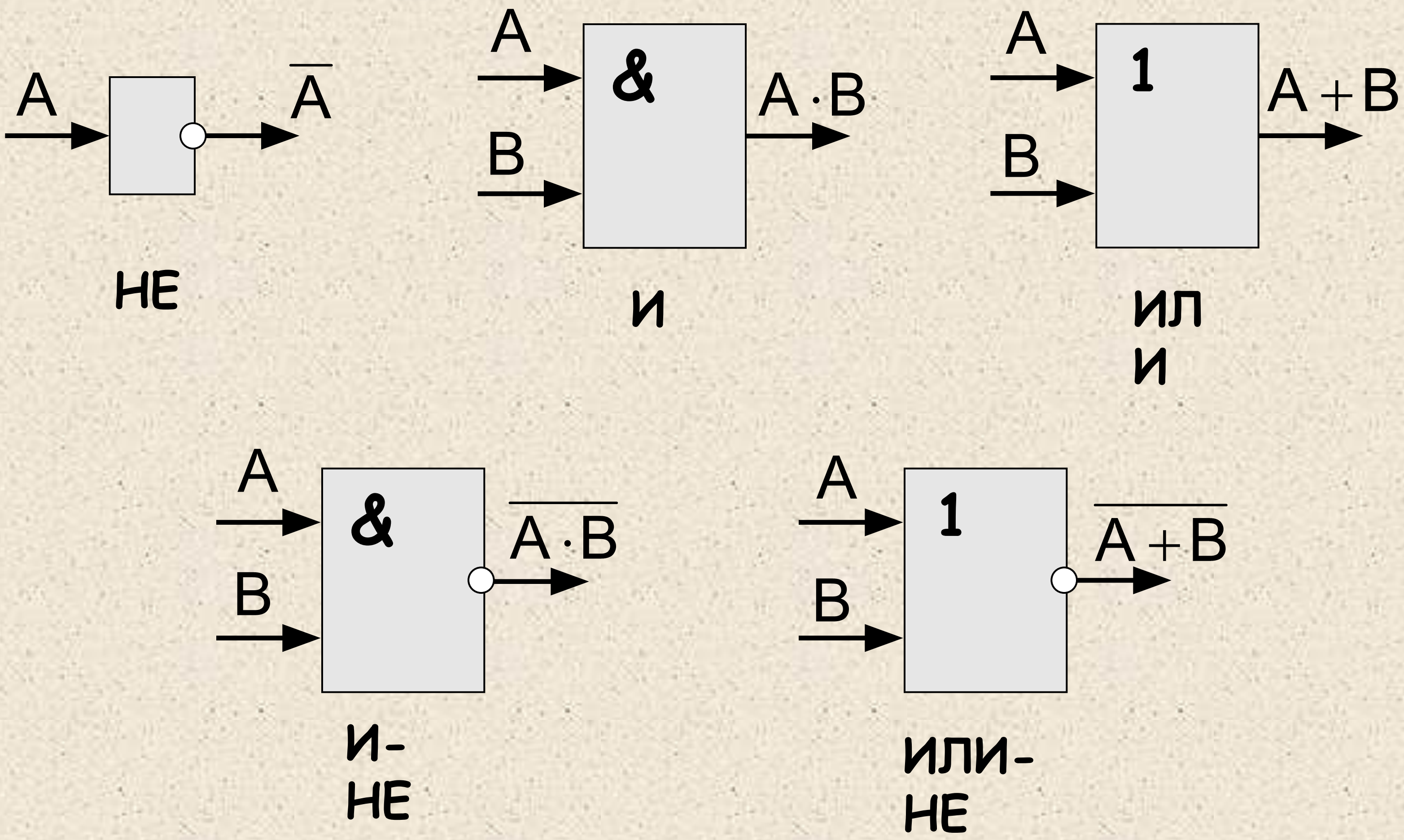
Логический элемент компьютера — это часть электронной логической схемы, которая реализует элементарную логическую функцию.

Логическими элементами компьютеров являются электронные схемы И, ИЛИ, НЕ, И—НЕ, ИЛИ—НЕ и другие.

Каждый логический элемент имеет свое условное обозначение, которое выражает его логическую функцию, но не указывает на то, какая именно электронная схема в нем реализована. Это упрощает запись и понимание сложных логических схем.

Логические элементы компьютера

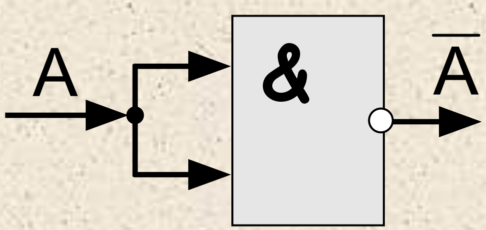
значок инверсии



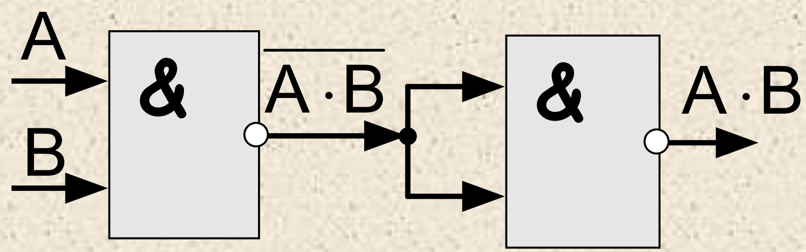
Логические элементы компьютера

Любое логическое выражение можно реализовать на элементах **И-НЕ** или **ИЛИ-НЕ**.

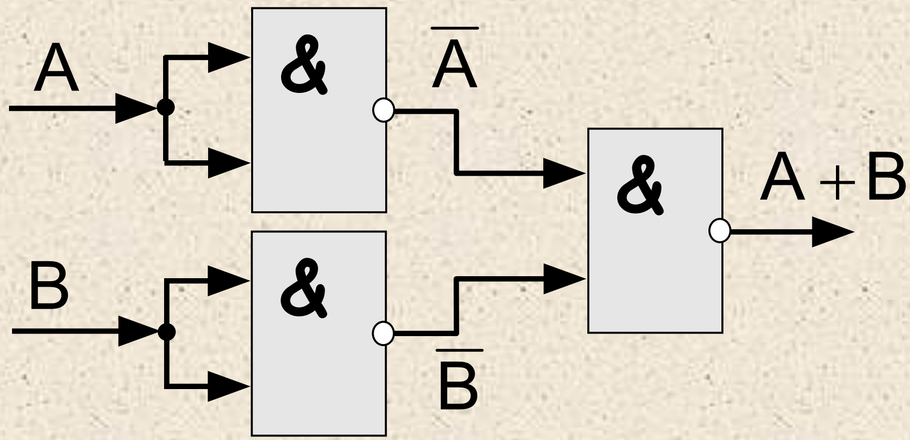
НЕ: $\overline{A} = \overline{A} + \overline{A} = \overline{A \cdot A}$



И: $A \cdot B = \overline{\overline{A \cdot B}}$



ИЛИ: $A + B = \overline{\overline{A} \cdot \overline{B}}$

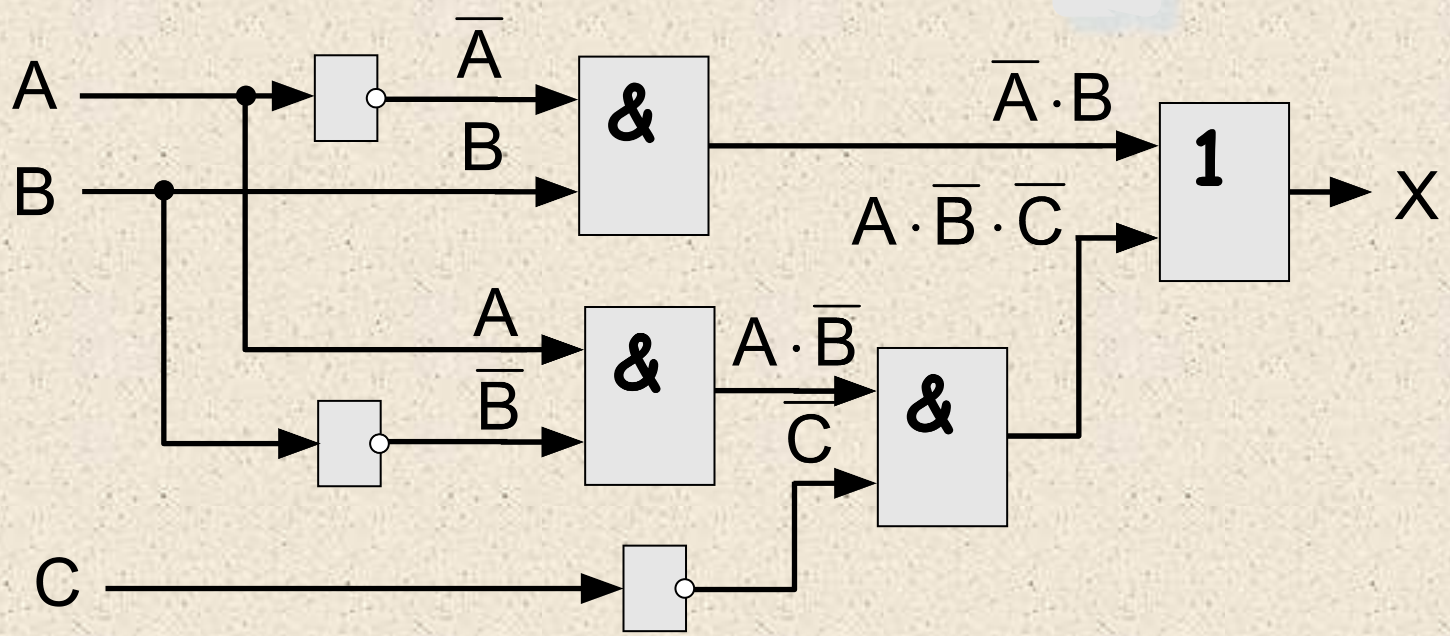


Составление схем

последняя операция - ИЛИ

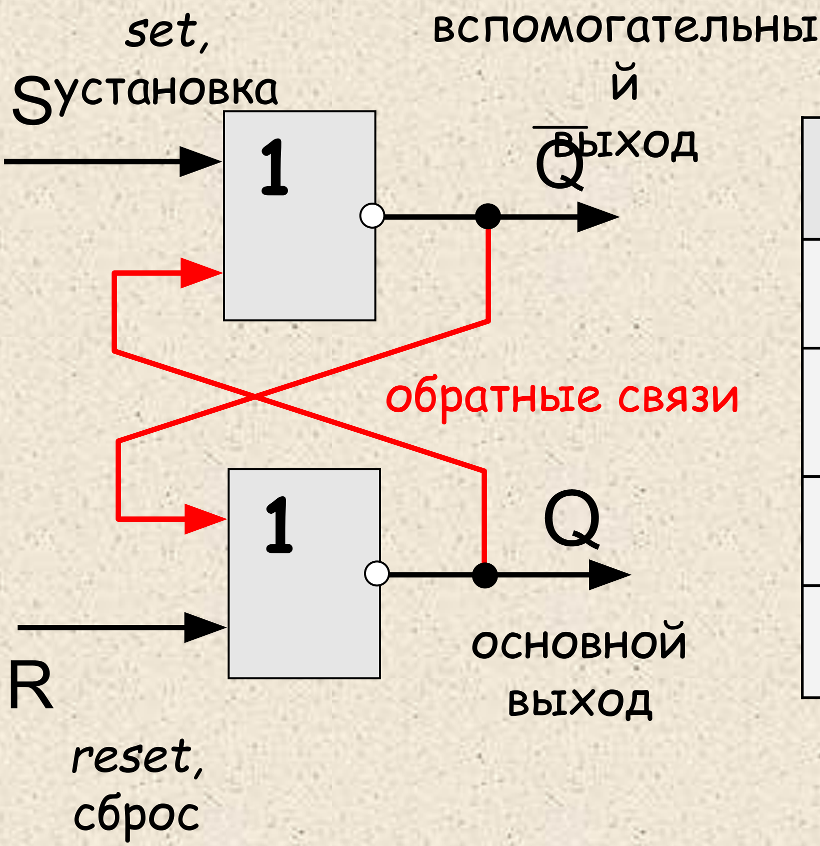
$$X = \overline{A} \cdot B + A \cdot \overline{B} \cdot \overline{C}$$

И



Триггер (англ. trigger – защёлка)

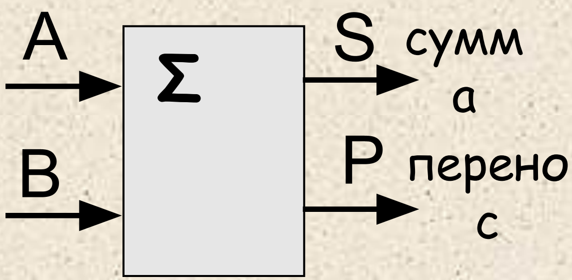
– это логическая схема, способная хранить 1 бит информации (1 или 0). Строится на 2-х элементах **ИЛИ-НЕ** или на 2-х элементах **И-НЕ**.



S	R	Q	$\overline{Q}$	режим
0	0	Q	$\overline{Q}$	хранение
0	1	0	1	сброс
1	0	1	0	установка 1
1	1	0	0	запрещен

Полусумматор

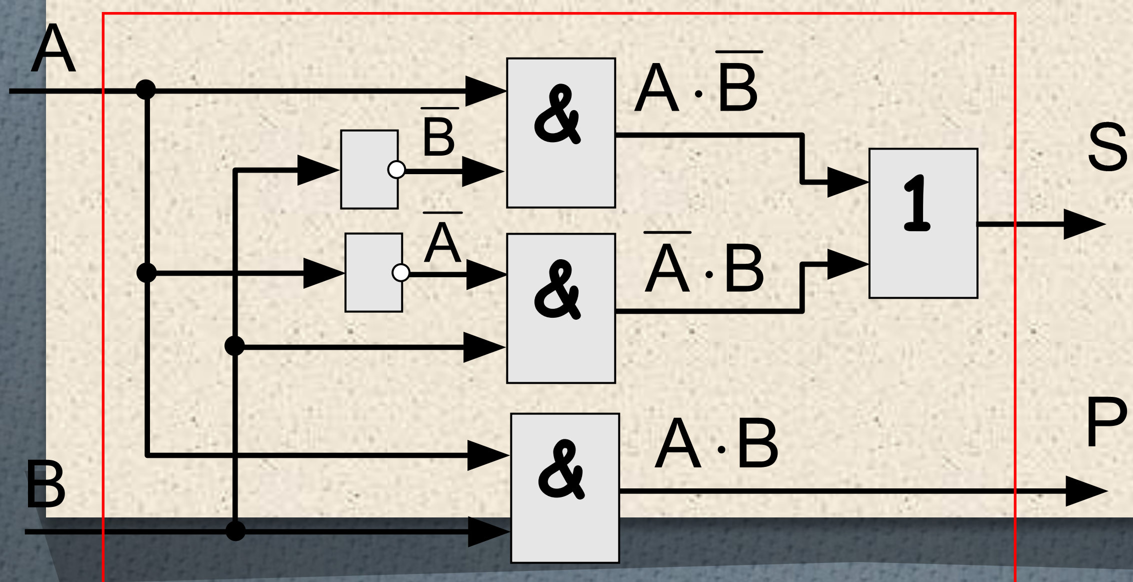
– это логическая схема, способная складывать два одноразрядных двоичных числа.



A	B	P	S
0	0	0 0	
0	1	0 1	
1	0	0 1	
1	1	1 0	

$P = A \cdot B$

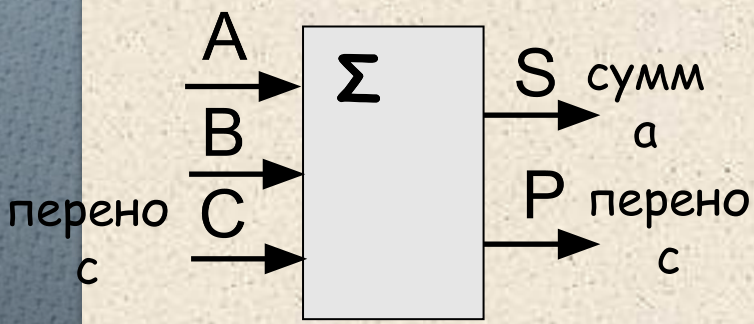
$S = A \oplus B = A \cdot \overline{B} + \overline{A} \cdot B$



$S = A \cdot \overline{B} + \overline{A} \cdot B$

Сумматор

- это логическая схема, способная складывать два одноразрядных двоичных числа с переносом из предыдущего разряда.



A	B	C	P	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1