Информатика

Институт информатики, инноваций и бизнес-систем

Кафедра информатики, инженерной и компьютерной графики

Черкасова Евгения Анатольевна

Тема 9. Логические основы компьютеров

Логические основы компьютеров

1. Логические выражения и операции
2. Преобразование логических выражений
3. Логические элементы компьютера

ВГУЭС

1 Логические выражения и операции

ВГУЭС

Булева алгебра

Двоичное кодирование – все виды информации кодируются с помощью 0 и 1.

Задача – разработать оптимальные правила обработки таких данных.

Джордж Буль разработал основы алгебры, в которой используются только 0 и 1 (алгебра логики, булева алгебра).

Почему "логика"? Результат выполнения операции можно представить как истинность (1) или ложность (0) некоторого высказывания.

Логические 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?

Обозначение высказываний

A – Сейчас идет дождь.
B – Форточка открыта.

простые высказывания (элементарные)

! **Любое высказывание может быть ложно (0) или истинно (1).**

Составные высказывания строятся из простых с помощью логических связок (операций) **"и"**, **"или"**, **"не"**, **"если … то"**, **"тогда и только тогда"** и др.

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

Операция НЕ (инверсия)

Если высказывание **A** истинно, то "**не A**" ложно, и наоборот.

A	не A
0	1
1	0

также: $\overline{\mathbf{A}}$,
not A (Паскаль),
! A (Си)

таблица истинности операции НЕ

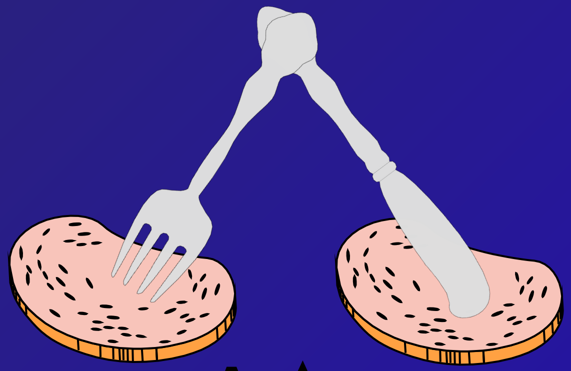
Таблица истинности логического выражения X – это таблица, где в левой части записываются все возможные комбинации значений исходных данных, а в правой – значение выражения X для каждой комбинации.

Операция И (логическое умножение, конъюнкция)

Высказывание "**А и В**" истинно тогда и только тогда, когда **А** и **В** истинны одновременно.

также: **A·B**, **A ∧ B**, **A and B** (Паскаль), **A && B** (Си)

	A	B	А и B
0	0	0	0
1	0	1	0
2	1	0	0
3	1	1	1



A ∧ B

конъюнкция – от лат. *conjunctio* — соединение

Операция ИЛИ (логическое сложение, дизъюнкция)

Высказывание "**А или В**" истинно тогда, когда истинно **А** или **В**, или оба вместе.

также: **A+B**, **A ∨ B**, **A or B** (Паскаль), **A || B** (Си)

A	B	A или B
0	0	0
0	1	1
1	0	1
1	1	1

дизъюнкция – от лат. *disjunctio* — разъединение

ВГУЭС

Операция "исключающее ИЛИ"

Высказывание "**A ⊕ B**" истинно тогда, когда истинно **A** или **B**, но ***не оба одновременно***.

также:
A xor B (Паскаль),
A ^ B (Си)

A	B	A ⊕ B
0	0	0
0	1	1
1	0	1
1	1	0

Свойства операции "исключающее ИЛИ"

$A \oplus 0 = A$ $\qquad$ $A \oplus A = 0$

$A \oplus 1 = \overline{A}$ $\qquad$ $(A \oplus B) \oplus B = ?$

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

A	B	$A\overline{B}$	$\overline{A}B$	$A\overline{B} + \overline{A}B$	$A \oplus B$
0	0	0	0	0	0
0	1	0	1	1	1
1	0	1	0	1	1
1	1	0	0	0	0

Импликация ("если …, то …")

Высказывание "**A → B**" истинно, если не исключено, что из **A** следует **B**.

A – "Работник хорошо работает".

B – "У работника хорошая зарплата".

A	B	A → B
0	0	1
0	1	1
1	0	0
1	1	1

$$A \rightarrow B = \overline{A} + B$$

Эквиваленция ("тогда и только тогда, …")

Высказывание "**A** ↔ **B**" истинно тогда и только тогда, когда **A** и **B** равны.

A	B	A ↔ B
0	0	1
0	1	0
1	0	0
1	1	1

$$A \leftrightarrow B = \overline{A \oplus B} = AB + \overline{A}\,\overline{B}$$

Базовый набор операций

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

И **ИЛИ** **НЕ**

базовый набор операций

Логические формулы

Система имеет три датчика и может работать, если два из них исправны.

A – "Датчик № 1 неисправен".

B – "Датчик № 2 неисправен".

C – "Датчик № 3 неисправен".

Аварийный сигнал:

X – "Неисправны два датчика".

X – "Неисправны датчики № 1 **и** № 2" **или**

"Неисправны датчики № 1 **и** № 3" **или**

"Неисправны датчики № 2 **и** № 3".

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

логическая формула

ВГУЭС

Составление таблиц истинности

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

	A	B	$A \cdot B$	$\overline{A} \cdot B$	$\overline{B}$	X
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

Логические выражения могут быть:

- **тождественно истинными** (всегда 1, тавтология)
- **тождественно ложными** (всегда 0, противоречие)
- **вычислимыми** (зависят от исходных данных)

Составление таблиц истинности

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

	A	B	C	AB	AC	BC	X
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

2 Преобразование логических выражений

Законы алгебры логики

название	для И	для ИЛИ
двойного отрицания	$\overline{\overline{A}} = A$	
исключения третьего	$A \cdot \overline{A} = 0$	$A + \overline{A} = 1$
операции с константами	$A \cdot 0 = 0, \quad A \cdot 1 = A$	$A + 0 = A, \quad A + 1 = 1$
повторения	$A \cdot A = A$	$A + A = A$
поглощения	$A \cdot (A + B) = A$	$A + A \cdot B = A$
переместительный	$A \cdot B = B \cdot A$	$A + B = B + A$
сочетательный	$A \cdot (B \cdot C) = (A \cdot B) \cdot C$	$A + (B + C) = (A + B) + C$
распределительный	$A + B \cdot C = (A + B) \cdot (A + C)$	$A \cdot (B + C) = A \cdot B + A \cdot C$
правила де Моргана	$\overline{A \cdot B} = \overline{A} + \overline{B}$	$\overline{A + B} = \overline{A} \cdot \overline{B}$

Упрощение логических выражений

Шаг 1. Заменить операции ⊕→↔ на их выражения через **И**, **ИЛИ** и **НЕ**:

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

$$A \rightarrow B = \overline{A} + B$$

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

Шаг 2. Раскрыть инверсию сложных выражений по формулам де Моргана:

$$\overline{A \cdot B} = \overline{A} + \overline{B}, \quad \overline{A + B} = \overline{A} \cdot \overline{B}$$

Шаг 3. Используя законы логики, упрощать выражение, стараясь применять закон исключения третьего.

ВГУЭС

Упрощение логических выражений

$$Q = M \cdot X \cdot \overline{H} + \overline{M} \cdot X \cdot \overline{H} = (M + \overline{M}) \cdot X \cdot \overline{H} = X \cdot \overline{H}$$

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

раскрыли →

$$= (\overline{B} + A) \cdot \overline{(A + B)} \cdot (\overline{A} + C)$$

формула де Моргана

$$= (\overline{B} + A) \cdot \overline{A} \cdot \overline{B} \cdot (\overline{A} + C)$$

распределительный

$$= (\overline{B} \cdot \overline{A} + A \cdot \overline{A}) \cdot \overline{B} \cdot (\overline{A} + C)$$

исключения третьего

$$= \overline{B} \cdot \overline{A} \cdot \overline{B} \cdot (\overline{A} + C)$$

повторения

$$= \overline{B} \cdot \overline{A} \cdot (\overline{A} + C)$$

поглощения

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

3 Логические элементы компьютера

ВГУЭС

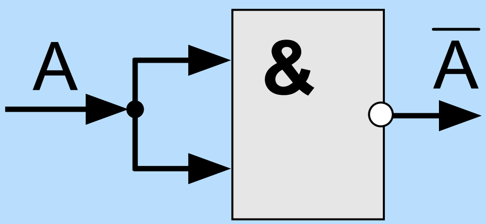
Логические элементы компьютера

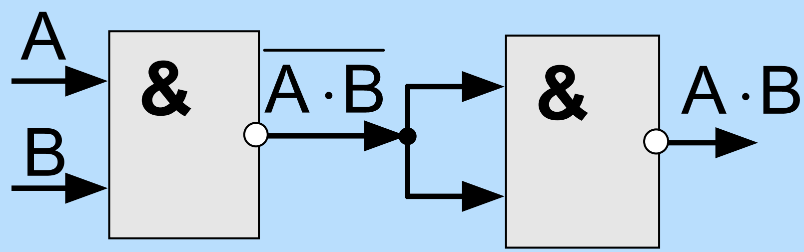
Логические элементы компьютера

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

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

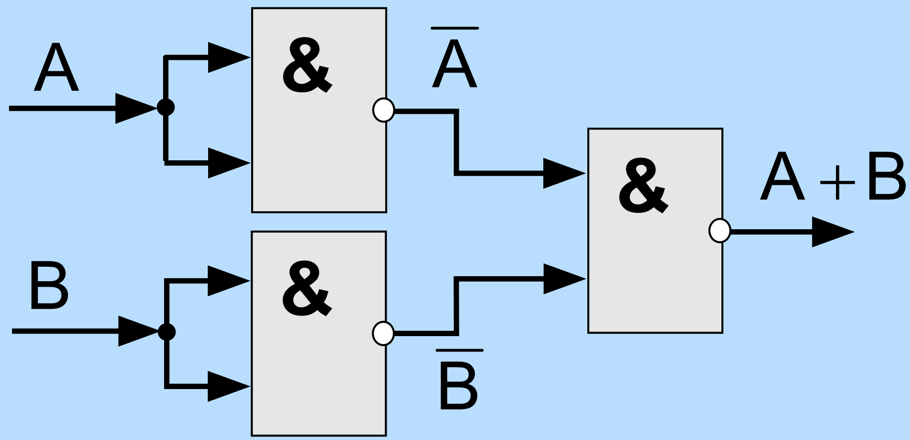
И: $A \cdot B = \overline{\overline{A \cdot B}}$





ИЛИ:

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

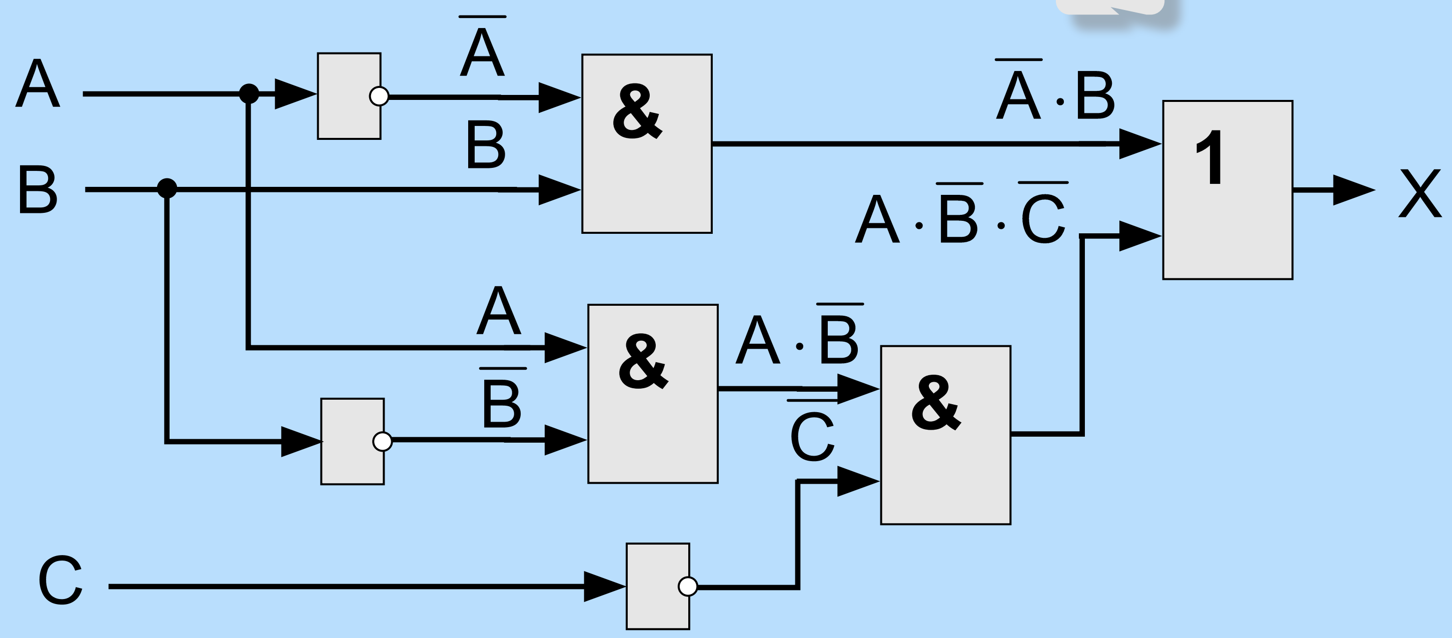


Составление схем

последняя операция - ИЛИ

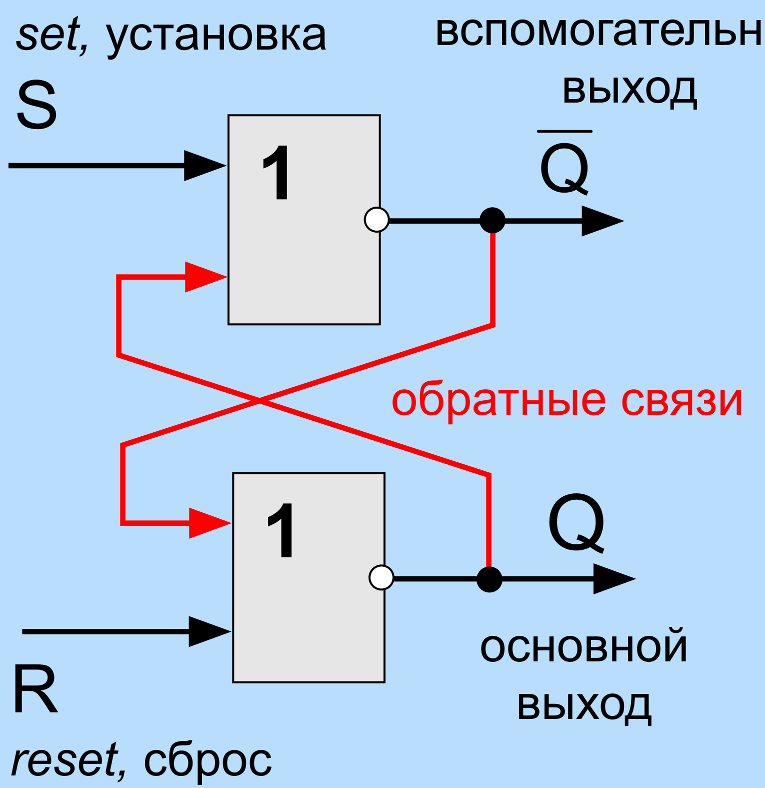
$$X = \overline{A} \cdot B + A \cdot \overline{B} \cdot \overline{C}$$

И



Триггер (англ. *trigger* – защёлка)

Триггер – 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.**

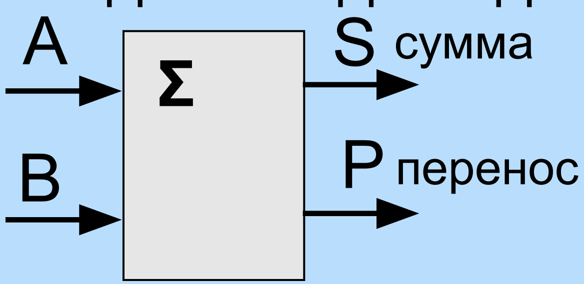


S	R	Q	$\overline{Q}$	режим
0	0	Q	$\overline{Q}$	хранение
0	1	0	1	сброс
1	0	1	0	установка 1
1	1	0	0	запрещен

ВГУЭС

Полусумматор

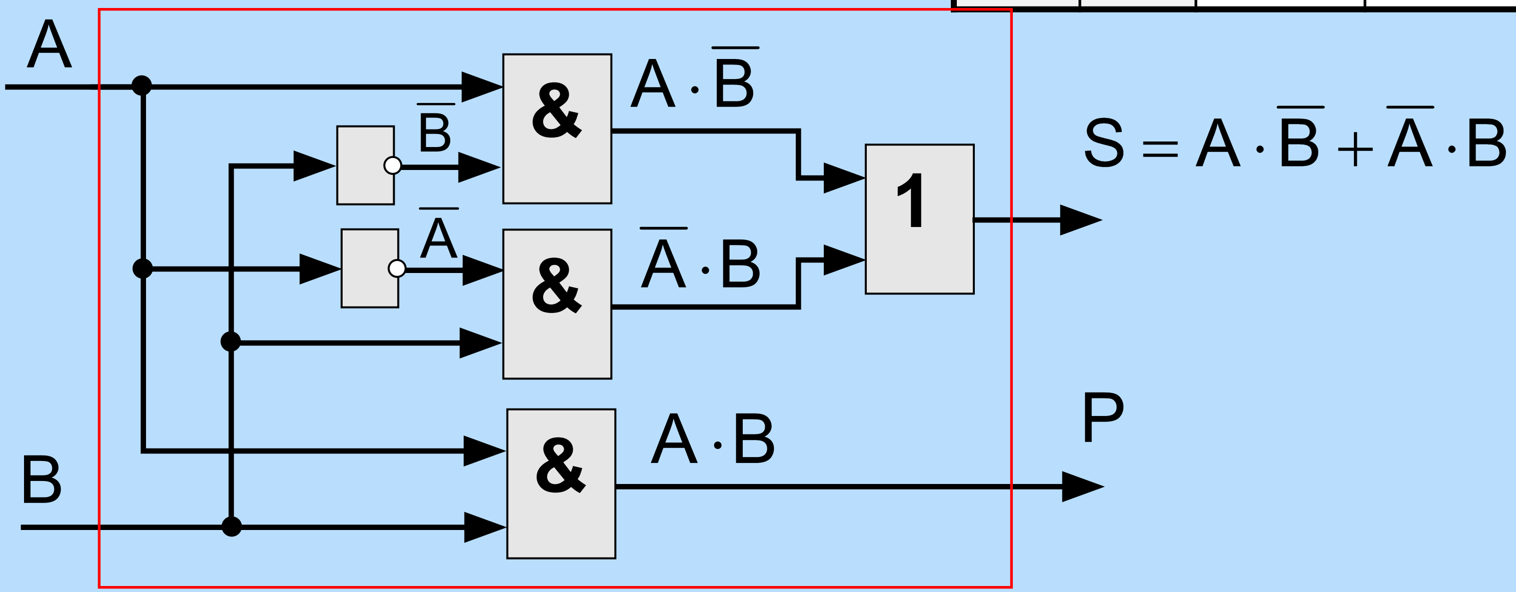
Полусумматор – это логическая схема, способная складывать два одноразрядных двоичных числа.



$$P = A \cdot B$$

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

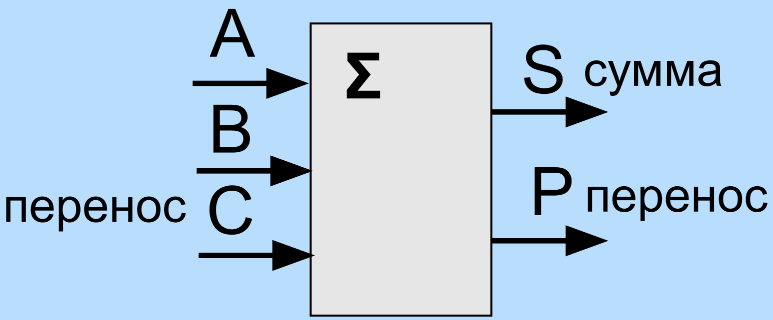
A	B	P	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0



ВГУЭС

Сумматор

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

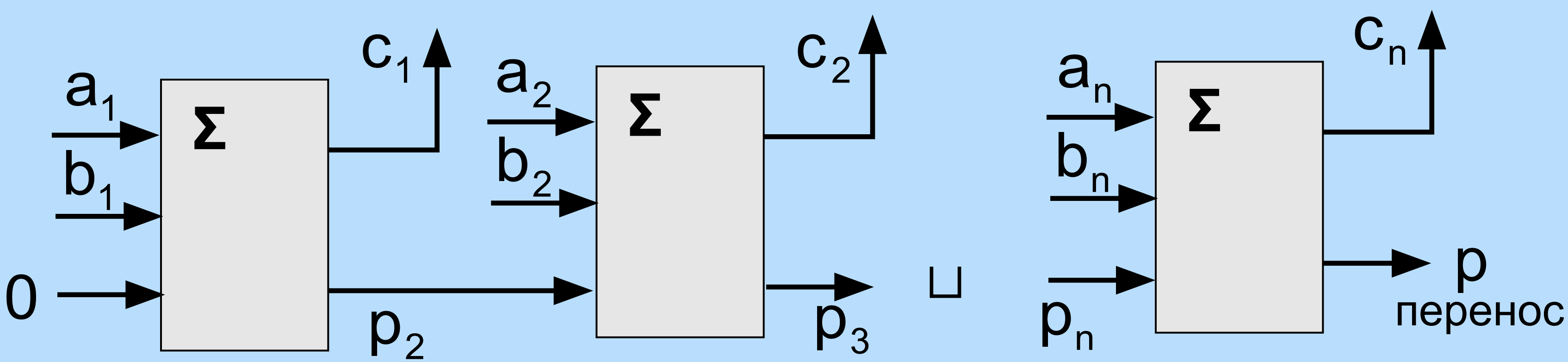
ВГУЭС

Многоразрядный сумматор

это логическая схема, способная складывать два n-разрядных двоичных числа.

$$
\begin{array}{rll}
 & A = & a_n \; a_{n-1} \; \ldots \; a_1 \\
+ & B = & b_n \; b_{n-1} \; \ldots \; b_1 \\
\hline
 & C = & \boxed{p} \; c_n \; c_{n-1} \; \ldots \; c_1
\end{array}
$$

перенос



ВГУЭС

Вопросы

Вопрос 1

Как записывается десятичное число 11 в двоичной системе счисления?

А) 1111

Б) 1101

В) 1011

Г) 1001

Вопрос 2

Операционная система – это ...

А) программа, обеспечивающая управление базами данных

Б) антивирусная программа

В) программа, управляющая работой компьютера

Г) система программирования

Вопрос 3

Какие пары объектов находятся в отношении "объект - модель"?

А) компьютер - данные

Б) компьютер - его функциональная схема

В) компьютер - программа

г) компьютер - алгоритм

Вопрос 4

Задан полный путь к файлу C:\DOC\PROBA.TXT Каково расширение файла, определяющее его тип?

А) C:\DOC\PROBA.TXT

Б) DOC\PROBA.TXT

В) PROBA.TXT

Г) TXT

ВГУЭС

Использование материалов презентации

Использование данной презентации, может осуществляться только при условии соблюдения требований законов РФ об авторском праве и интеллектуальной собственности, а также с учетом требований настоящего Заявления.

Презентация является собственностью авторов. Разрешается распечатывать копию любой части презентации для личного некоммерческого использования, однако не допускается распечатывать какую-либо часть презентации с любой иной целью или по каким-либо причинам вносить изменения в любую часть презентации. Использование любой части презентации в другом произведении, как в печатной, электронной, так и иной форме, а также использование любой части презентации в другой презентации посредством ссылки или иным образом допускается только после получения письменного согласия авторов.

ВГУЭС