Архитектура компьютера.

Биты и манипулирование ими.

«Но да будет слово ваше: да, да; нет, нет; а что сверх того, то от лукавого.»

- Евангелие от Матфея 5, 37

Компьютер решает задачи в соответствие с алгоритмом, представленным в виде *машинного кода* – последовательности нулей и единиц.

Бит – двоичный разряд, имеющий два значения – нуль или единицу.
Теоретическим основанием для технических реализаций систем, манипулирующих битами является *булева алгебра* (или изоморфные ей математические структуры – *алгебра высказываний* и *алгебра логики*).

На множестве из двух элементов – 0 и 1 (или «правда» и «ложь», или «да» и «*нет*») заданы две бинарные операции – конъюнкция *and* и дизъюнкция *or*, и одна унарная – *not*.

Свойства логических операций:

закон двойного отрицания: *not not a = a*

***закон коммутативности*:**

a or b = b or a

a and b = b and a

***закон ассоциативности*:**

a or (b or c) = (a or b) or c

a and (b and c) = (a and b) and c

***закон дистрибутивности*:**

a or (b and c) = (a or b) and (a or c)

a and (b or c) = (a and b) or (a and c)

***правила де Моргана*:**

not (a or b) = not a and not b

not (a and b) = not a or not b

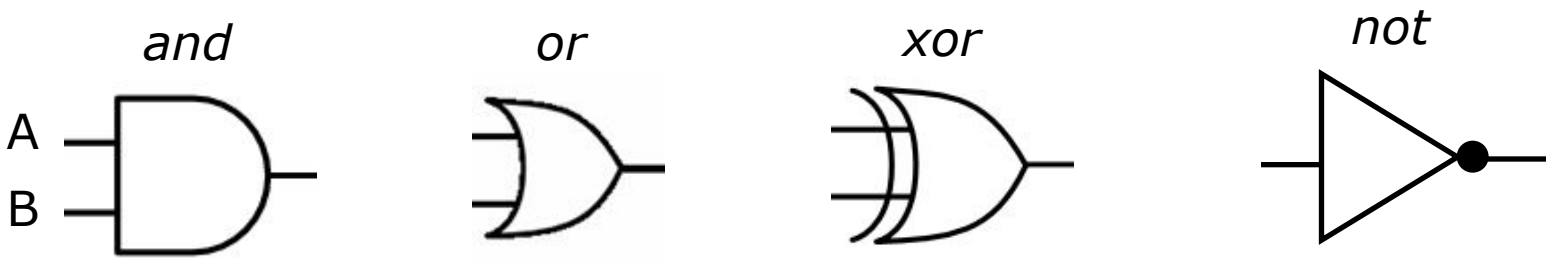
Дополнительная операция – «*исключающее или*» *xor*

Таблица истинности:

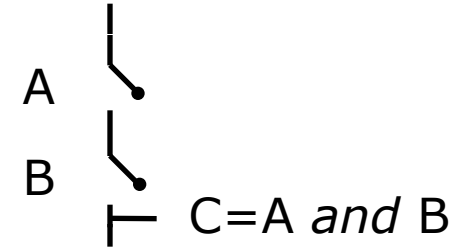
a	b	*and*	*or*	*xor*
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

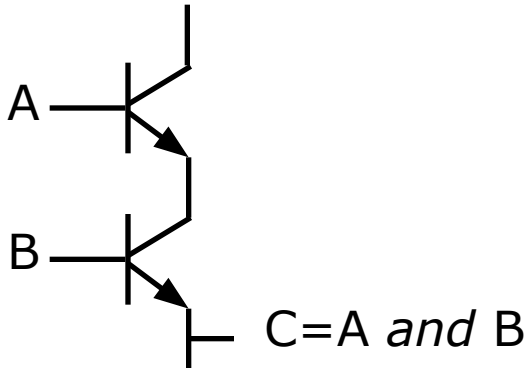
a	*not*
0	*1*
1	*0*

Абстрактные устройства, реализующие логические операции (*вентили*):



Техническая реализация вентилей:





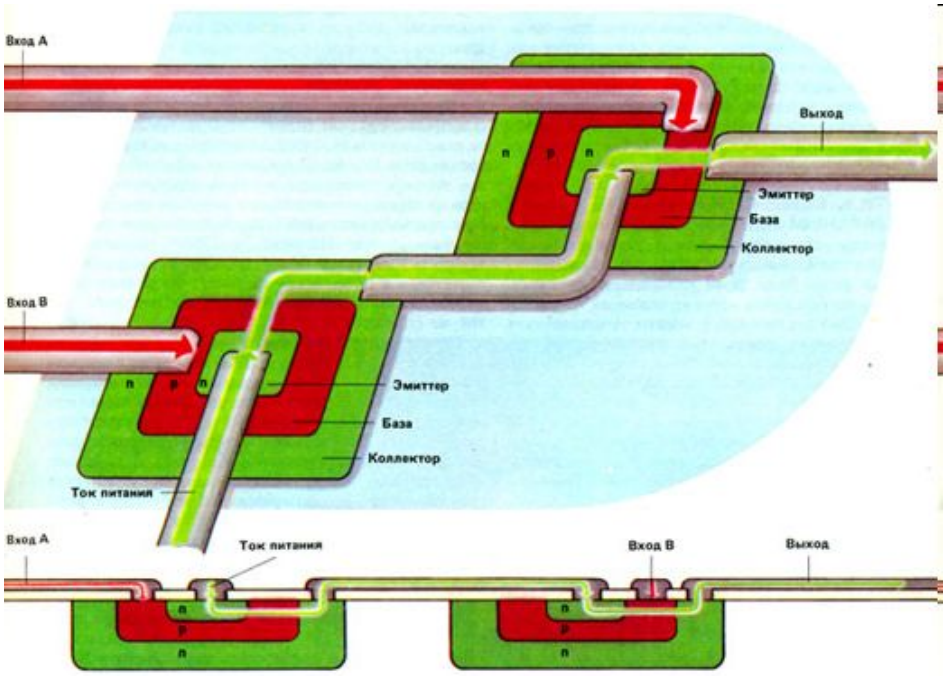


Таблица истинности:

A	B	F
0	***0***	***1***
0	1	0
1	***0***	***1***
1	***1***	***1***

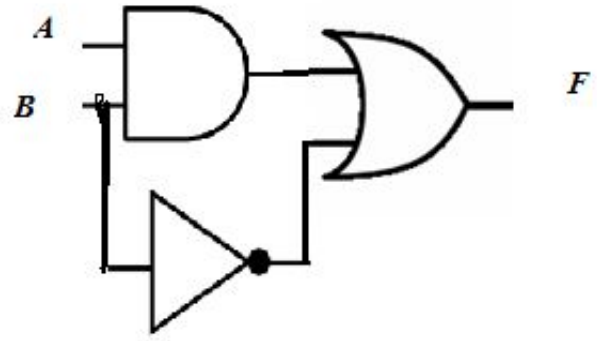
Представление таблицы истинности логической функцией:

$$F = \overline{A}\,\overline{B} + A\overline{B} + AB$$

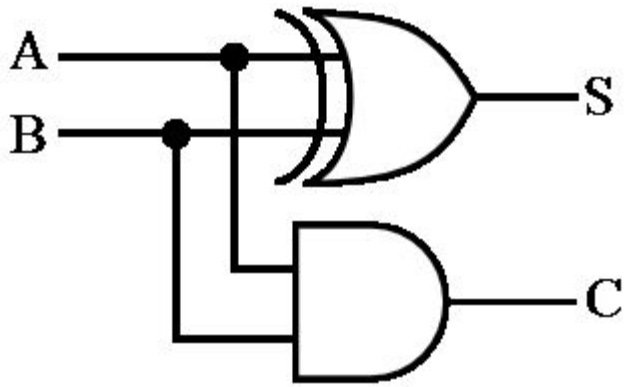
Упрощение логического выражения:

$$F = \overline{A}\,\overline{B} + A\overline{B} + AB = \overline{B}(A + \overline{A}) + AB = \overline{B} + AB.$$

Схема соответствующего абстрактного устройства:

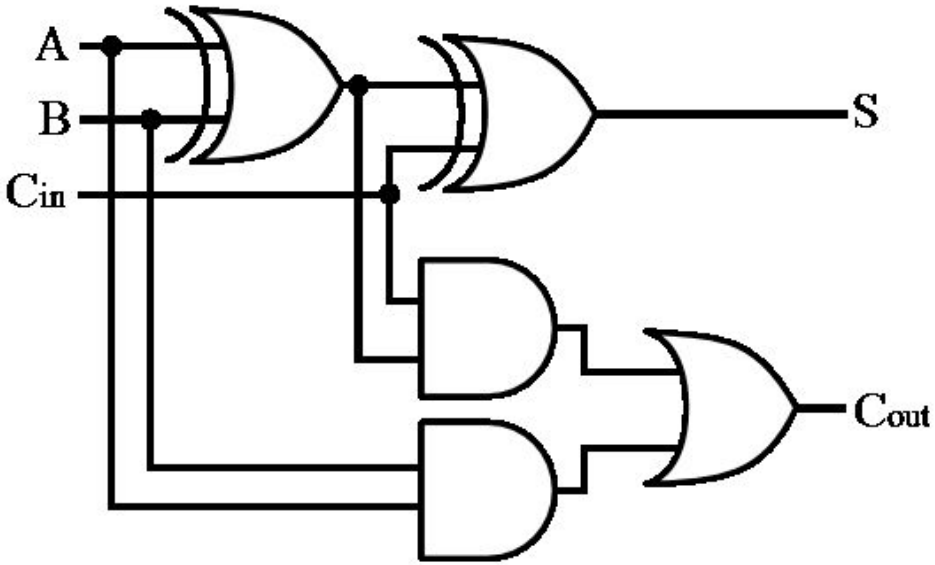


Двоичный полусумматор:



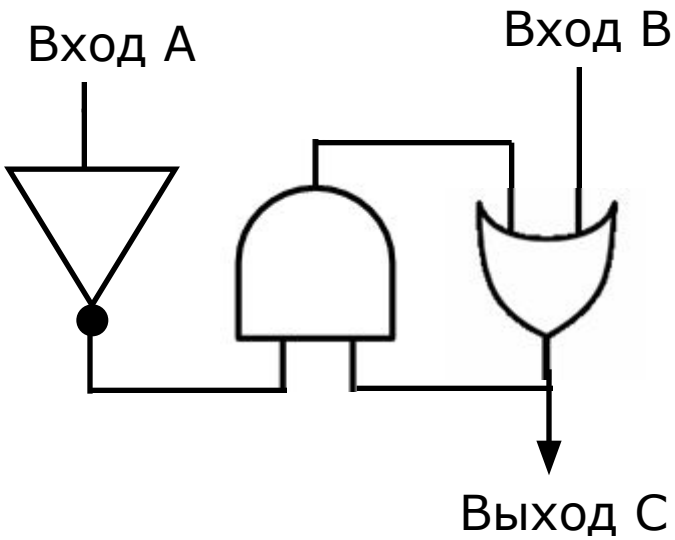
A	B	S	C
0	0	0	0
1	0	1	0
0	1	1	0
1	1	0	1

Полный двоичный сумматор:

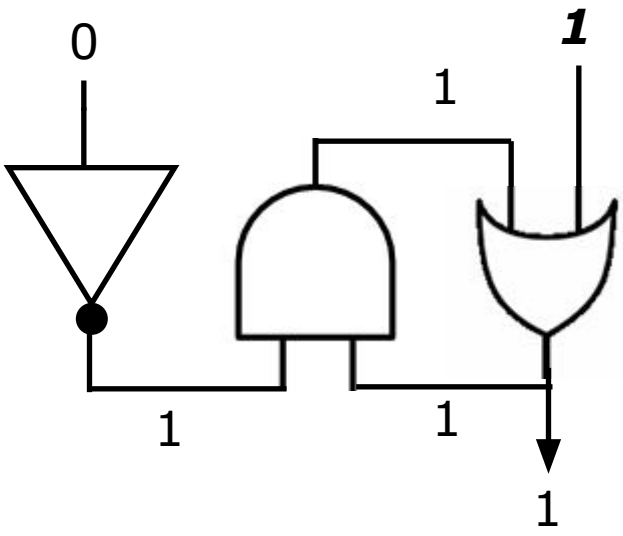


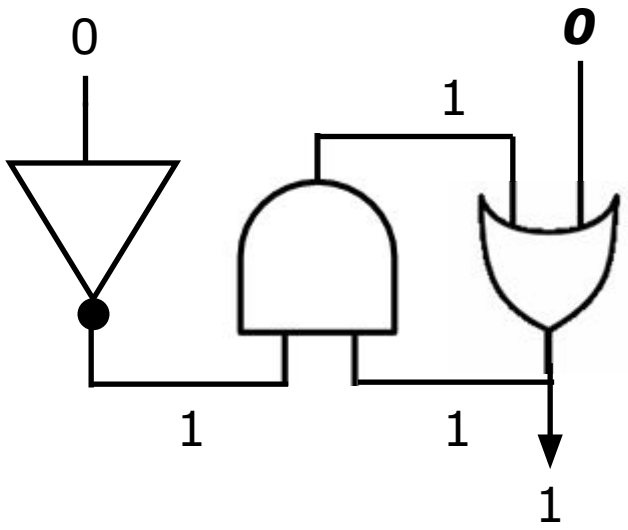
A	B	Cin	S	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
1	0	0	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

Триггер:



Подача сигнала на вход В устанавливает триггер в состояние 1. После снятия напряжения с этого входа триггер остается в этом состоянии. Для перехода триггера в состояние 0 необходимо подать сигнал на вход А.

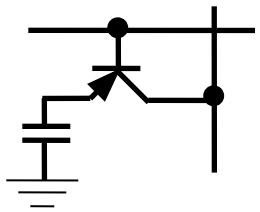




Техническая реализация триггера дорогостоящая, поэтому биты с помощью триггеров хранят в небольшой по объему памяти, но с большим быстродействием. Эта статическая память – *SRAM*, используется в персональных компьютерах для *регистров* и *кэшей*.

Оперативная память основа на технологии динамической памяти – DRAM, использующей конденсаторы для хранения бит.

Элемент DRAM:



И его схематическое изображение: