

## Лекция №3

# СХЕМОТЕХНИКА ВЫЧИСЛИТЕЛЬНЫХ УСТРОЙСТВ

Дисциплина: Архитектура вычислительных систем и компьютерные сети

Преподаватель: Миронов Константин Валерьевич

Поток: ПРО-3

Учебный год: 2024/25



# Цифровая схемотехника



Существует 2 типа электронных схем:

- **Комбинационные** – состояние выходов зависит от состояния входов в данный момент
  - Надо учитывать, что каждого элемента есть время задержки, обусловленное неидеальностью технологии изготовления, температурой.
- **Последовательностные** (схемы с памятью) - <...> а также от предшествующего состояния

# Содержание лекции



- Комбинационные схемы
  - **Синтез универсальных комбинационных схем с помощью карт Карно**
  - (Де)кодеры и (де)мультиплексоры
- Последовательностные схемы
  - Триггеры
  - Регистры и счетчики
- Аппаратная реализация арифметических операций
  - Сумматоры и вычитатели
  - Умножители

# Синтез комбинационных схем

## Методика

- Входные данные – таблица истинности
- Таблица истинности преобразуется в логическое выражение
  - Дизъюнктивные и конъюнктивные нормальные формы (ДНФ и КНФ)
    - *Совершенные* ДНФ и КНФ
    - Для функций от небольшого числа переменных можно использовать **карты Карно** для построения *минимальных* ДНФ или КНФ
- На большой комбинационной схеме можно попробовать выразить одни функции или их части через другие
- Итоговая логическая функция реализуется в виде схемы на доступных логических элементах

# Синтез комбинационных схем

## СДНФ и СКНФ

$F =$	$X_1$	$X_2$	$X_3$	$X_4$	$F$	$F =$
$\bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4+$	0	0	0	0	1	$(x_1 + x_2 + x_3 + \bar{x}_4) *$
$\bar{x}_1\bar{x}_2x_3\bar{x}_4+$	0	0	0	1	0	$(x_1 + x_2 + \bar{x}_3 + \bar{x}_4) *$
$\bar{x}_1\bar{x}_2x_3x_4+$	0	0	1	0	1	$(x_1 + x_2 + \bar{x}_3 + \bar{x}_4) *$
$\bar{x}_1\bar{x}_2x_3x_4+$	0	0	1	1	0	$(x_1 + \bar{x}_2 + x_3 + \bar{x}_4) *$
$\bar{x}_1x_2\bar{x}_3\bar{x}_4+$	0	1	0	0	1	$(x_1 + \bar{x}_2 + x_3 + \bar{x}_4) *$
$\bar{x}_1x_2\bar{x}_3\bar{x}_4+$	0	1	0	1	0	$(x_1 + \bar{x}_2 + \bar{x}_3 + \bar{x}_4) *$
$\bar{x}_1x_2x_3\bar{x}_4+$	0	1	1	0	1	
$\bar{x}_1x_2x_3\bar{x}_4+$	0	1	1	1	0	
$x_1\bar{x}_2\bar{x}_3\bar{x}_4+$	1	0	0	0	1	$(\bar{x}_1 + x_2 + x_3 + \bar{x}_4) *$
$x_1\bar{x}_2\bar{x}_3\bar{x}_4+$	1	0	0	1	0	$(\bar{x}_1 + x_2 + \bar{x}_3 + \bar{x}_4) *$
$x_1\bar{x}_2x_3\bar{x}_4+$	1	0	1	0	1	$(\bar{x}_1 + x_2 + \bar{x}_3 + \bar{x}_4) *$
$x_1\bar{x}_2x_3\bar{x}_4+$	1	0	1	1	0	$(\bar{x}_1 + \bar{x}_2 + x_3 + x_4) *$
$x_1x_2\bar{x}_3\bar{x}_4+$	1	1	0	0	0	$(\bar{x}_1 + \bar{x}_2 + \bar{x}_3 + x_4) *$
$x_1x_2\bar{x}_3\bar{x}_4+$	1	1	0	1	1	
$x_1x_2x_3\bar{x}_4+$	1	1	1	0	0	
$x_1x_2x_3x_4$	1	1	1	1	1	

Выражения формируются тривиально, но неоптимальны с точки зрения количества логических элементов

# Синтез комбинационных схем

## Карты Карно

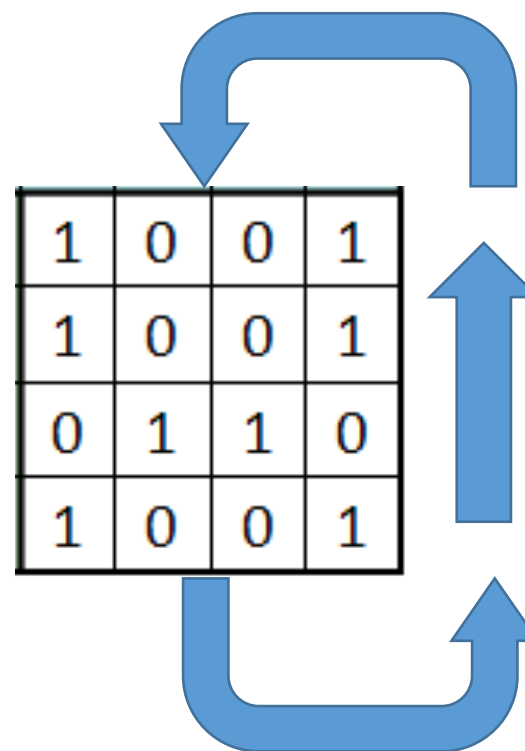
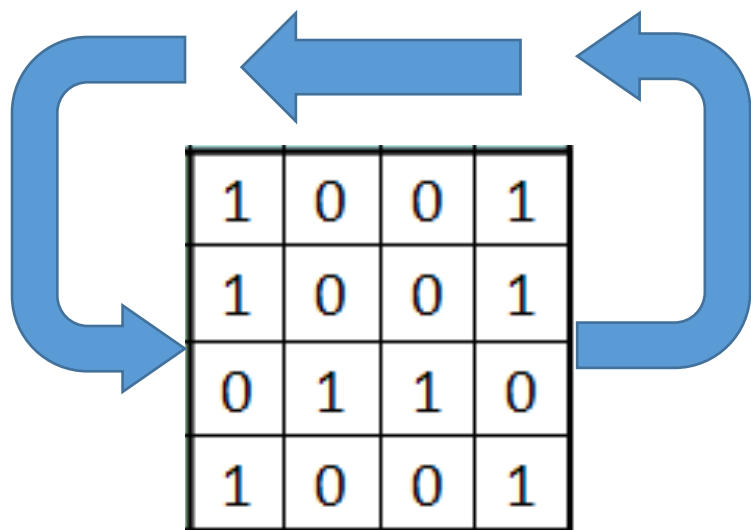
$X_1$	$X_2$	$X_3$	$X_4$	$F$
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1



$X_3 X_4$		00	01	11	10
$X_1 X_2$	00	1	0	0	1
	01	1	0	0	1
	11	0	1	1	0
	10	1	0	0	1

# Синтез комбинационных схем

## Карты Карно



Карта Карно закольцована по горизонтали и вертикали:  
После четвертой строки идет первая  
После четвертого столбца идет первый

# Синтез комбинационных схем

## Карты Карно

Компактная ДНФ строится путем объединения true-ячеек в группы

- Группы на карте Карно – это прямоугольники, у которых длина стороны – степень двойки (1,2,4 клетки)
- Группы строятся с учетом закольцованности карты
- Группы могут пересекаться
- Каждая группа должна быть как можно больше
- Количество групп должно быть как можно меньше

$X_3 X_4$		00	01	11	10
$X_1 X_2$	00	1	0	0	1
	01	1	0	0	1
	11	0	1	1	0
	10	1	0	0	1



# Синтез комбинационных схем

## Карты Карно

Минимальная ДНФ:

- Каждая группа – отдельное слагаемое
- В слагаемое записывается произведение тех переменных, которые имеют одно значение внутри группы
- Инверсии переменных в произведении определяются так же как в СДНФ

$$F = \bar{x}_1\bar{x}_4 + \bar{x}_2\bar{x}_4 + x_1x_2x_4$$

$X_3 X_4$		00	01	11	10
$X_1 X_2$	00	1	0	0	1
	01	1	0	0	1
	11	0	1	1	0
	10	1	0	0	1

# Синтез комбинационных схем

## Карты Карно

Минимальная КНФ строится аналогично

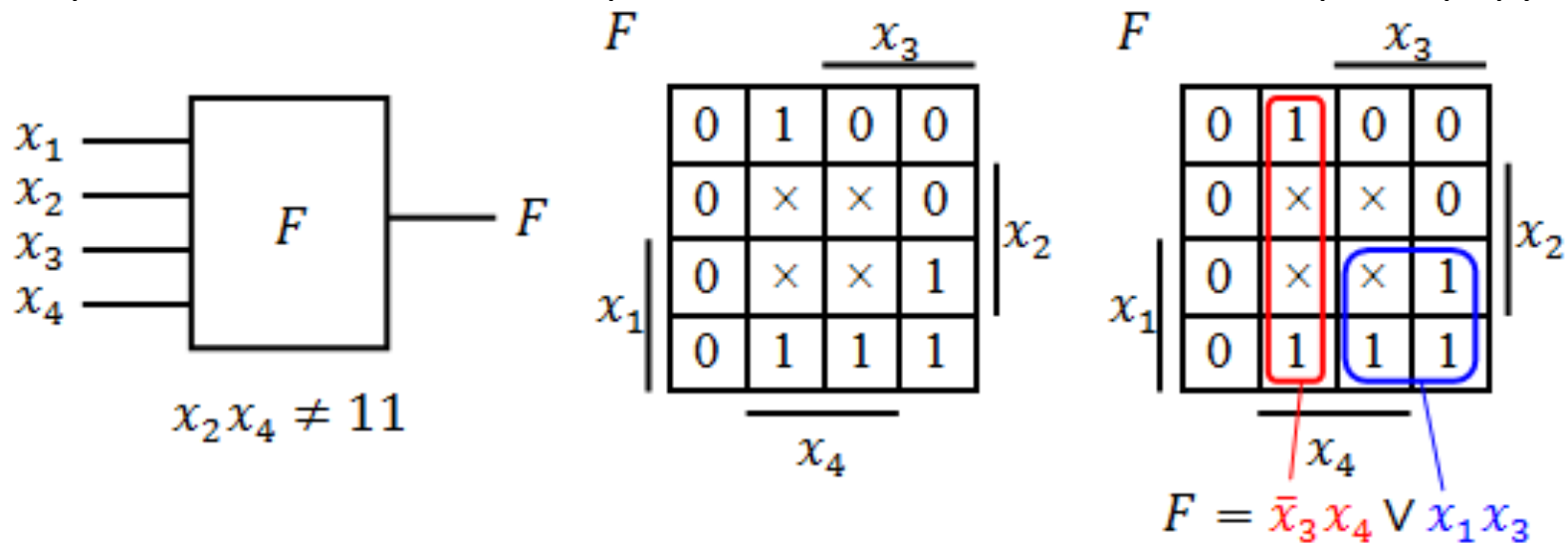
$X_3 X_4$		00	01	11	10
$X_1 X_2$	00	1	0	0	1
	01	1	0	0	1
	11	0	1	1	0
	10	1	0	0	1

$$F = (x_1 + \bar{x}_4)(x_2 + \bar{x}_4)(\bar{x}_1 + \bar{x}_2 + x_4)$$

# Синтез комбинационных схем

## Карты Карно: особенности

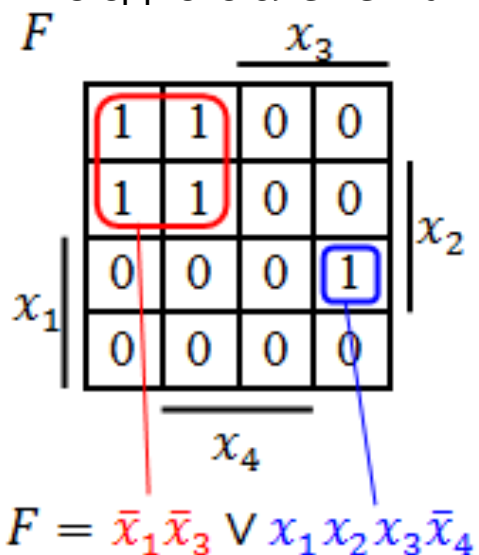
Часто возникает ситуация, когда некоторые значения таблицы истинности не важны для работы схемы. Тогда им выбирают такие значения, которые обеспечивают максимальный размер групп на карте Карно



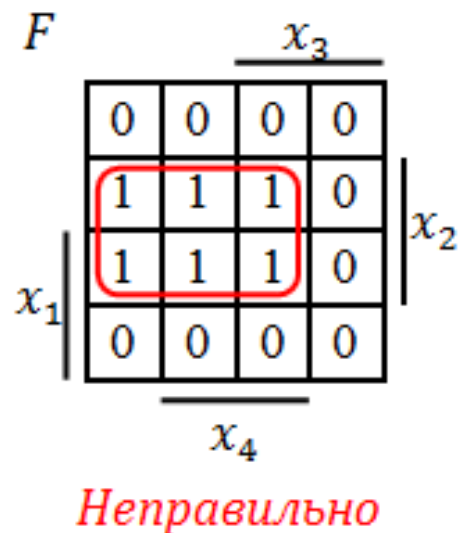
# Синтез комбинационных схем

## Карты Карно: особенности

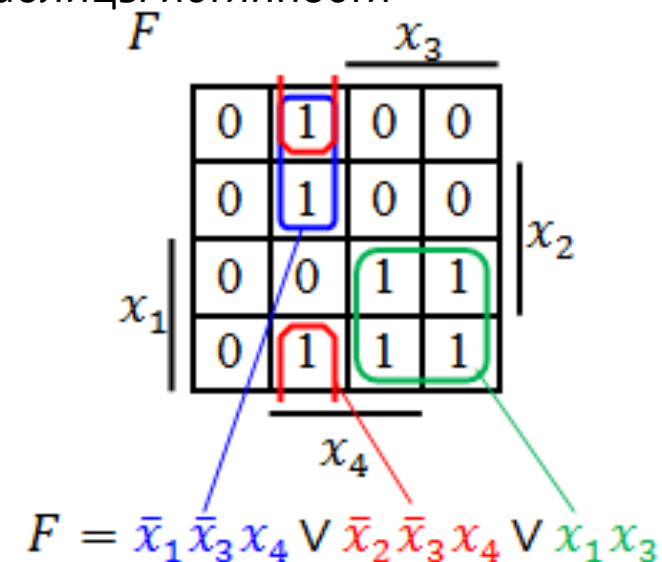
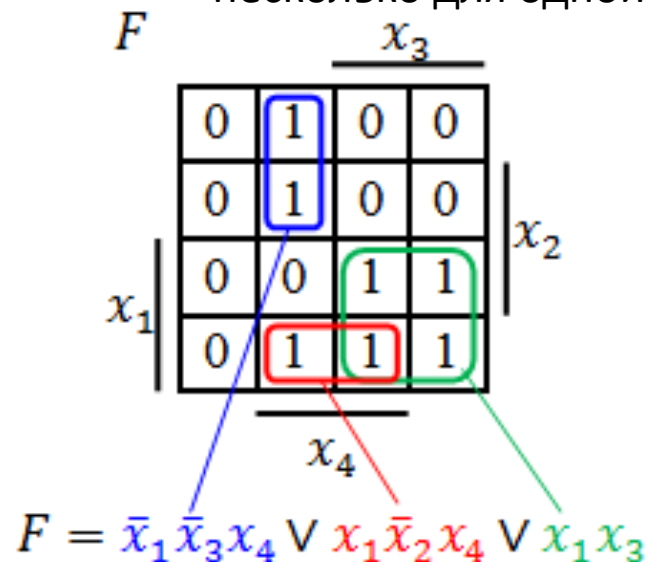
Группа может состоять из одного элемента



Сторона группы не может состоять из 3 клеток



Минимальных нормальных форм может быть несколько для одной таблицы истинности



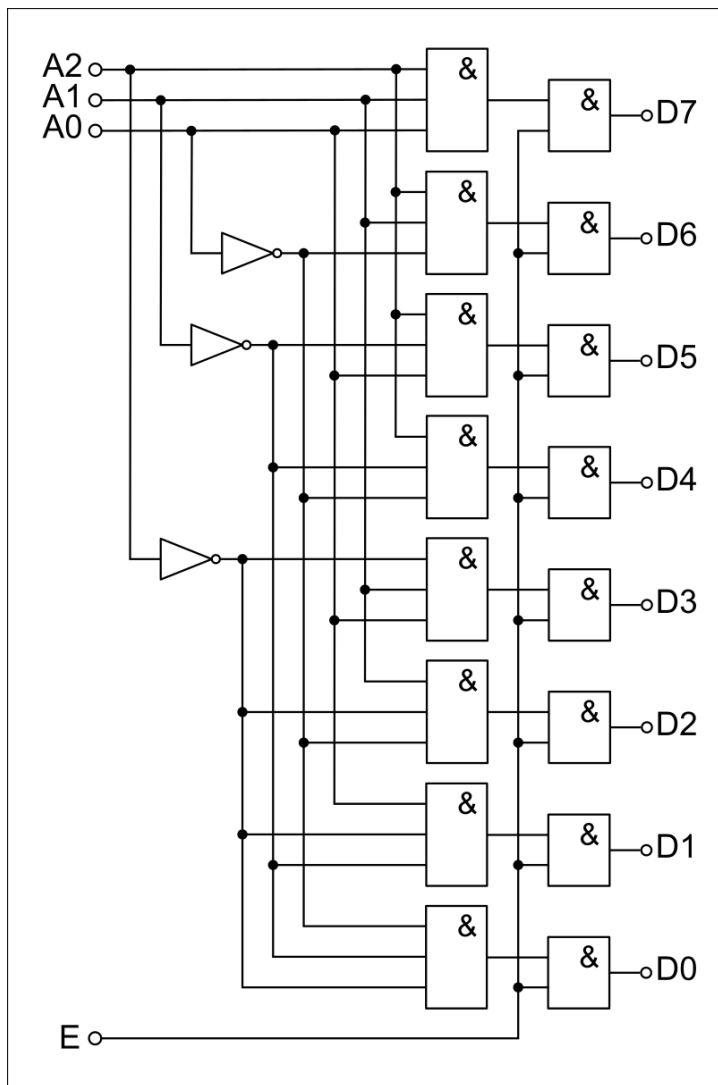
# Содержание лекции



- Комбинационные схемы
  - Синтез универсальных комбинационных схем с помощью карт Карно
  - **(Де)кодеры и (де)мультиплексоры**
- Последовательностные схемы
  - Триггеры
  - Регистры и счетчики
- Аппаратная реализация арифметических операций
  - Сумматоры и вычитатели
  - Умножители

# Комбинационные схемы

## Декодер (дешифратор)



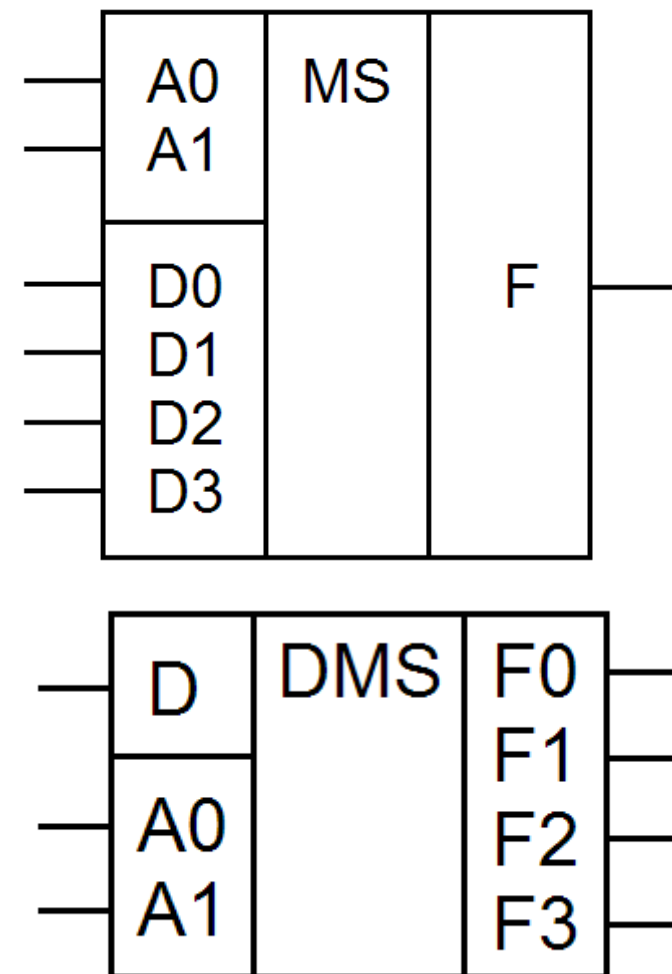
Адрес			Разрешение	Состояние выходов							
A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>	E	D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
0	0	0	0	x	x	x	x	x	x	x	x
0	0	0	1	0	0	0	0	0	0	0	1
0	0	1	0	x	x	x	x	x	x	x	x
0	0	1	1	0	0	0	0	0	0	1	0
0	1	0	0	x	x	x	x	x	x	x	x
0	1	0	1	0	0	0	0	0	1	0	0
0	1	1	0	x	x	x	x	x	x	x	x
0	1	1	1	0	0	0	0	1	0	0	0
1	0	0	0	x	x	x	x	x	x	x	x
1	0	0	1	0	0	0	1	0	0	0	0
1	0	1	0	x	x	x	x	x	x	x	x
1	0	1	1	0	0	1	0	0	0	0	0
1	1	0	0	x	x	x	x	x	x	x	x
1	1	0	1	0	1	0	0	0	0	0	0
1	1	1	0	x	x	x	x	x	x	x	x
1	1	1	1	1	0	0	0	0	0	0	0

Перевод из двоичного  
кода в унарный

# Комбинационные схемы

## Кодер, мультиплексор, демультиплексор

- **Кодер (шифратор)** – преобразование из унарного кода в двоичный
  - Реализуется через комбинацию логических сложений
- **Мультиплексор** – вывод сигналов от разных источников к одному получателю
  - А кодирует номер источника, с которого идет сигнал
  - Реализуется на основе декодера, выходы которого перемножаются с выходами соответствующих источников
  - Используется для вывода данных из регистров
  - Любую таблицу истинности можно реализовать через мультиплексор с 1 и 0 на источниках
- **Демультиплексор** – передача сигнала от одного источника разным получателям
  - Реализуется аналогично мультиплексору
  - Используется для записи данных в регистры



# Содержание лекции



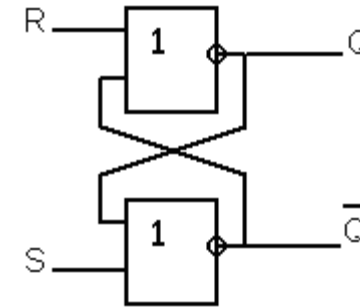
- Комбинационные схемы
  - Синтез универсальных комбинационных схем с помощью карт Карно
  - (Де)кодеры и (де)мультиплексоры
- Последовательностные схемы
  - **Триггеры**
  - Регистры и счетчики
- Аппаратная реализация арифметических операций
  - Сумматоры и вычитатели
  - Умножители



# Последовательные схемы

## Бистабильная ячейка

- Схема из двух стрелок Пирса или штрихов Шеффера, выходы которых перекрестно соединены с входами
- Может хранить 1 бит информации
- На основе бистабильных ячеек строятся **триггеры** [flip-flop] – устройства с двумя выходами (один – инверсия второго), позволяющие хранить 1 бит информации
- Сама по себе бистабильная ячейка может рассматриваться как **асинхронный RS-триггер**



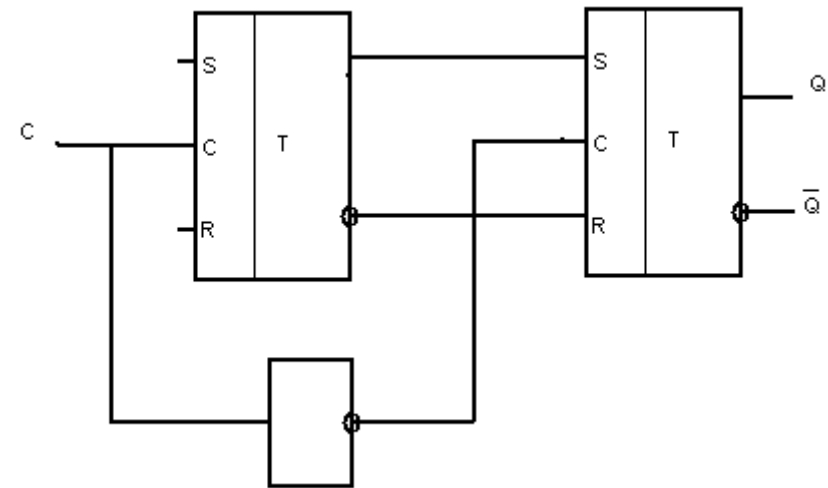
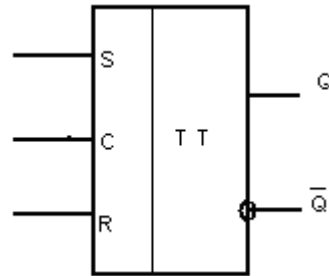
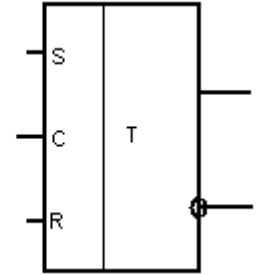
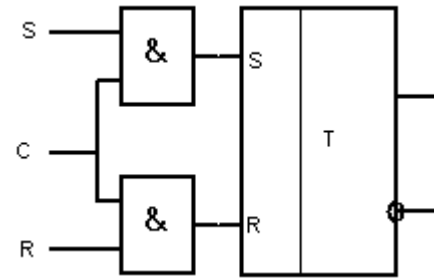
S	R	$Q_{n+1}$	Режим
0	0	$Q_n$	хранение
1	0	1	запись 1
0	1	0	запись 0
1	1	-	не допускается

На самом деле значение будет, но оба выхода окажутся равны

# Последовательные схемы

## Триггеры

- **Синхронный одноктактный RS-триггер** – переключается между состояниями по тактовому сигналу
- **Двухтактный RS-триггер** – двухтактная структура позволяет строить схемы, в которых можно не только хранить предыдущее состояние, но и инвертировать его

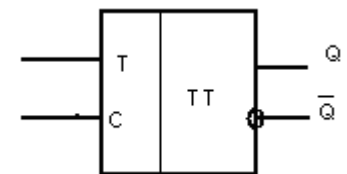
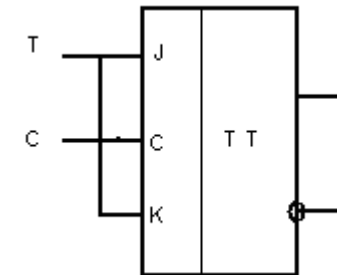
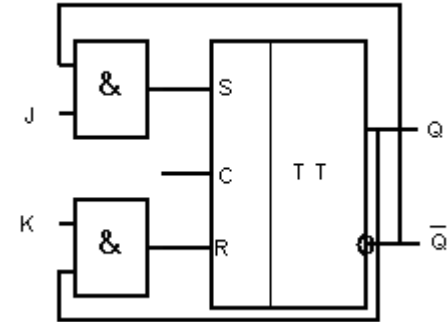


# Последовательные схемы

## Триггеры

- **JK-триггер** – универсальный триггер, при входных сигналах 11 (запретных для RS-триггера) инвертирует свое предыдущее значение
- **T-триггер** – JK-триггер, у которого на оба входа подается один сигнал
  - В таблице истинности остаются только первая и последняя строчки (хранение и инверсия)
  - Если синхросигнал колеблется с заданной частотой, то выход будет колебаться с вдвое меньшей
  - Если соединить T-триггеры в цепочку (Q предыдущего подается на T следующего), каждый следующий будет колебаться с вдвое меньшей частотой – получится **счетчик импульсов**

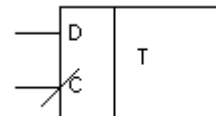
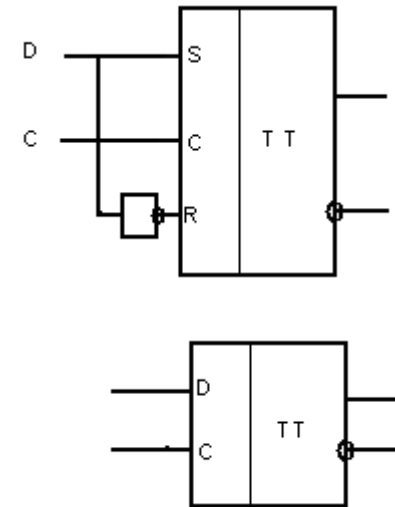
J	K	$Q_{n+1}$	Режим
0	0	$Q_n$	Хранение
1	0	1	Запись 1
0	1	0	Запись 0
1	1	$\overline{Q_n}$	инверсия



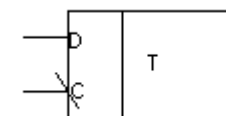
# Последовательные схемы

## Триггеры

- **D-триггер** – JK- или RS-триггер, у которого инверсия первого входа подается на второй
  - В таблице истинности остаются только вторая и третья строчки (0 и 1)
  - Обеспечивает задержку сигнала на 1 такт
  - Если соединить в цепочку D-триггеры (Q предыдущего подается на D следующего), то на каждом такте значение предыдущего триггера будет переходить на следующий – получится **сдвиговый регистр**
- **DV-** и **TV-**триггеры – варианты D- и T-триггеров, в которых есть дополнительный сигнал V (Verbot), запрещающий/разрешающий переключение состояний (как бы дополнительный синхросигнал)
- Триггеры с динамическим управлением – переключаются при изменении уровня синхросигнала



Изменение  
с 0 на 1



Изменение  
с 1 на 0

# Содержание лекции

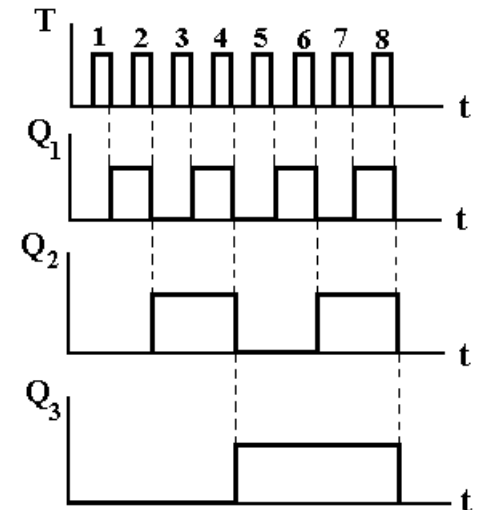
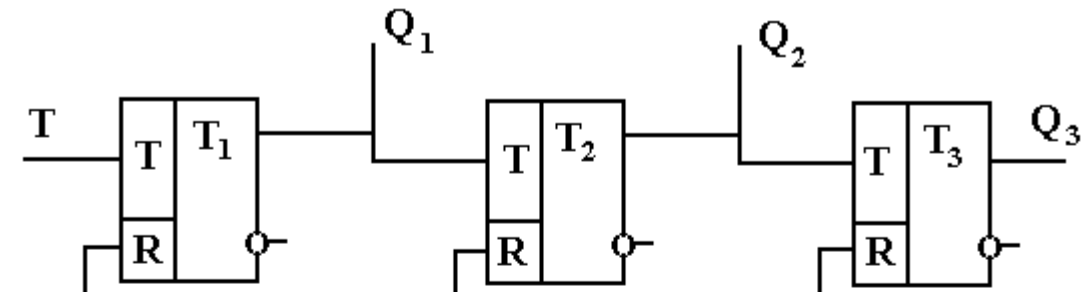


- Комбинационные схемы
  - Синтез универсальных комбинационных схем с помощью карт Карно
  - (Де)кодеры и (де)мультиплексоры
- Последовательностные схемы
  - Триггеры
  - **Регистры и счетчики**
- Аппаратная реализация арифметических операций
  - Сумматоры и вычитатели
  - Умножители

# Последовательные схемы

## Счетчики импульсов

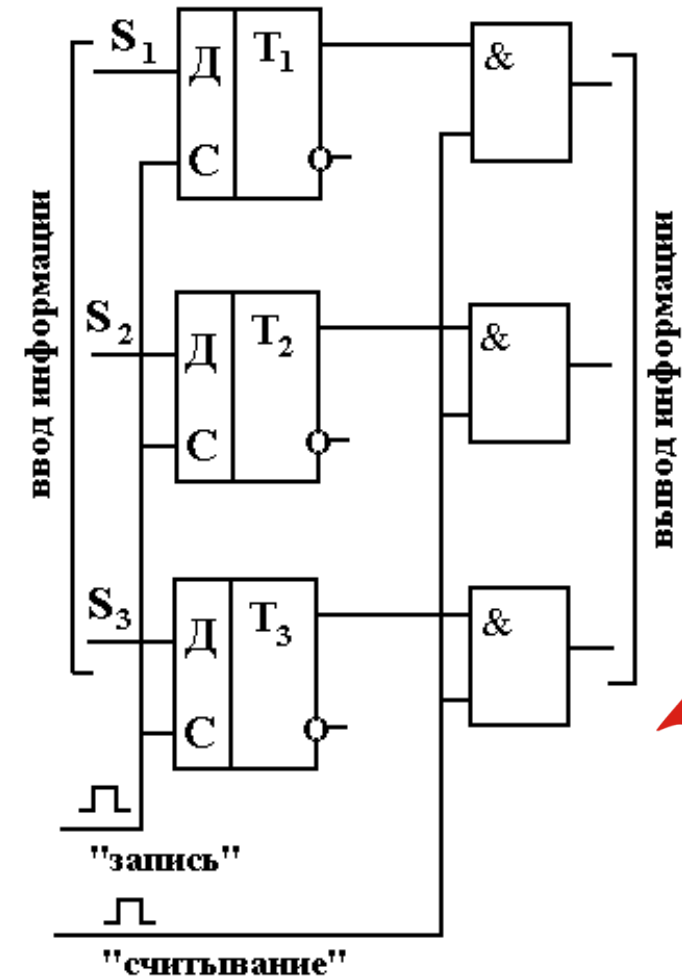
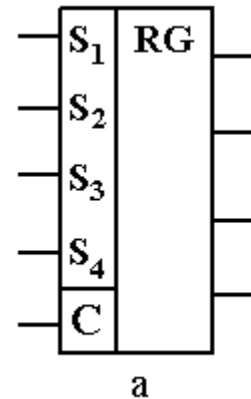
- Варианты:
  - Цепочка Т-триггеров (Q предыдущего подается на Т следующего)
  - Цепочка JK-триггеров (Q предыдущего подается на J и K следующего)
- Вход К JK-триггеров позволяет обнулить значение счетчика
  - \* на картинке используется вход R RS-триггера, на котором построен Т-триггер
- Если читать разряды Q справа налево, получится число тактов, прошедших с прошлого обнуления
- Аналогично на триггерах можно строить другие счетчики (вычитающие, в специальных кодах, с заданным модулем)



# Последовательные схемы

## Регистры

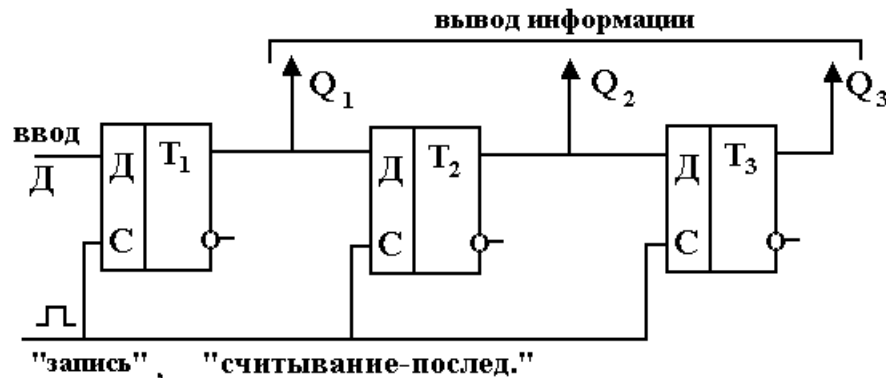
- Массив триггеров для хранения многоразрядных данных
- Типы:
  - Параллельный ввод и вывод
  - Сдвиговые регистры:
    - Последовательный ввод и/или вывод



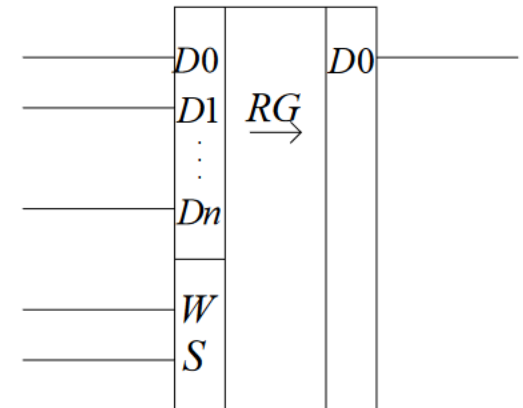
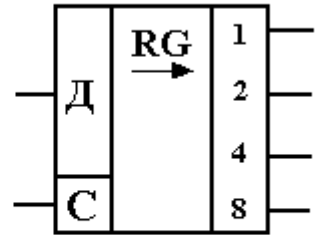
# Последовательностные схемы

## Сдвиговые регистры

- Преобразуют последовательный код в параллельный и/или наоборот
- Варианты:
  - Цепочка D-триггеров (Q предыдущего подается на D следующего)
  - Цепочка JK/RS-триггеров (Q предыдущего подается на J/S следующего, а  $\text{not}(Q)$  – на K/R)



Преобразование последовательного  
кода в параллельный





# Содержание лекции



- Комбинационные схемы
  - Синтез универсальных комбинационных схем с помощью карт Карно
  - (Де)кодеры и (де)мультиплексоры
- Последовательностные схемы
  - Триггеры
  - Регистры и счетчики
- Аппаратная реализация арифметических операций
  - **Сумматоры и вычитатели**
  - Умножители

# Комбинационные схемы

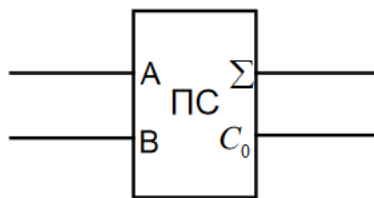
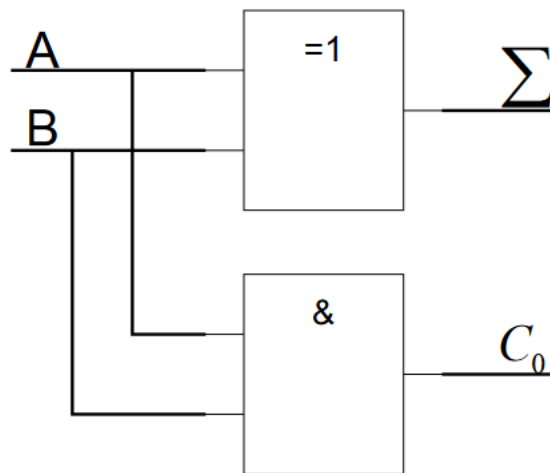
## Полусумматор

- это устройство, предназначенное для осуществления операции арифметического сложения двух одноразрядных чисел.

A	B	$\Sigma$	$C_0$
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$\Sigma = A \oplus B$$

$$C_0 = A * B$$

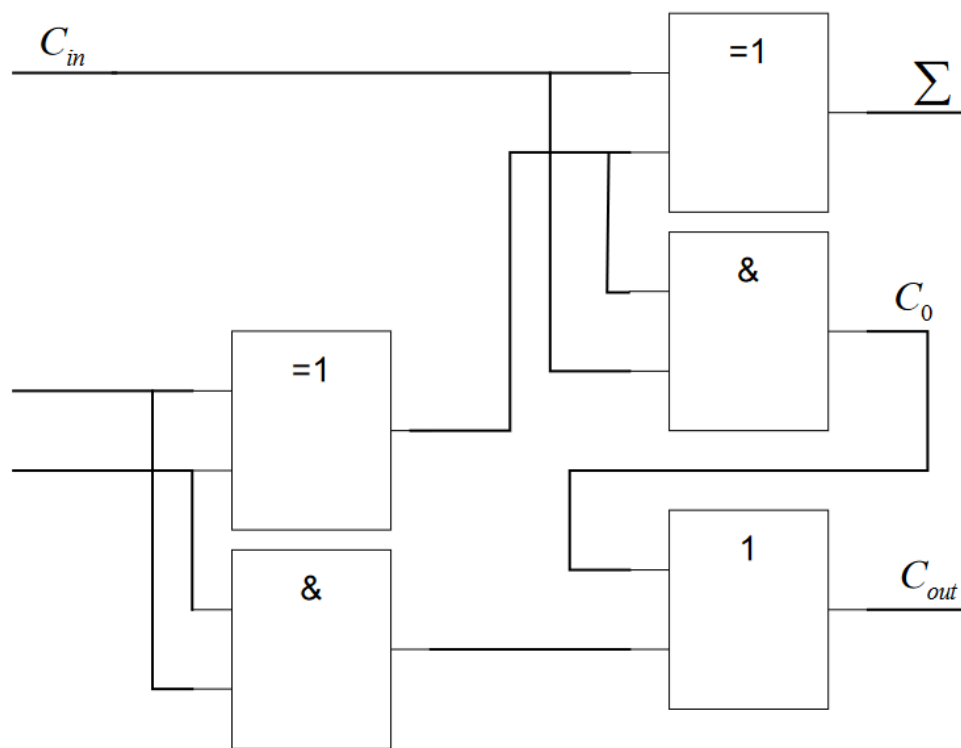


# Комбинационные схемы

## Полный сумматор

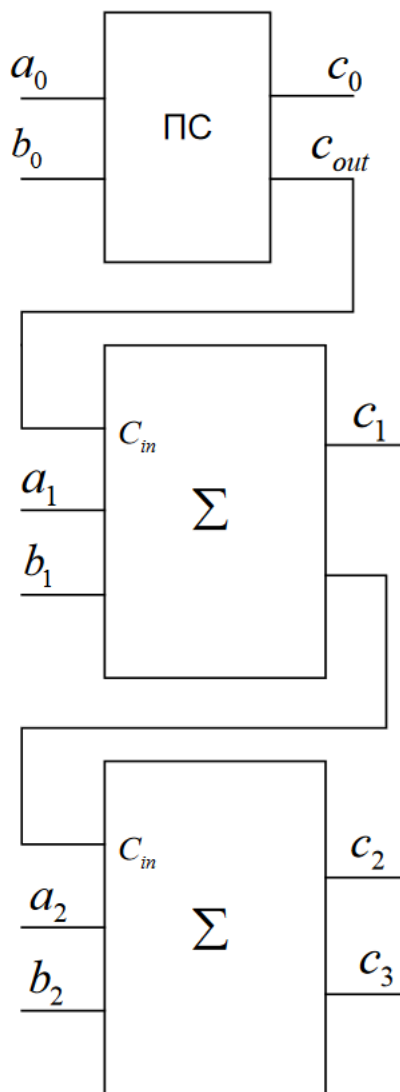
- это устройство, предназначенное для арифметического сложения двух одноразрядных чисел с учетом переноса из младшего разряда.

A	B	$C_{in}$	$\Sigma$	$C_{out}$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1
1	1	1	0	1



# Комбинационные схемы

## Многоразрядные сумматоры (параллельные)



$$C = (A + B) \bmod 2^n$$

$n$  – число разрядов

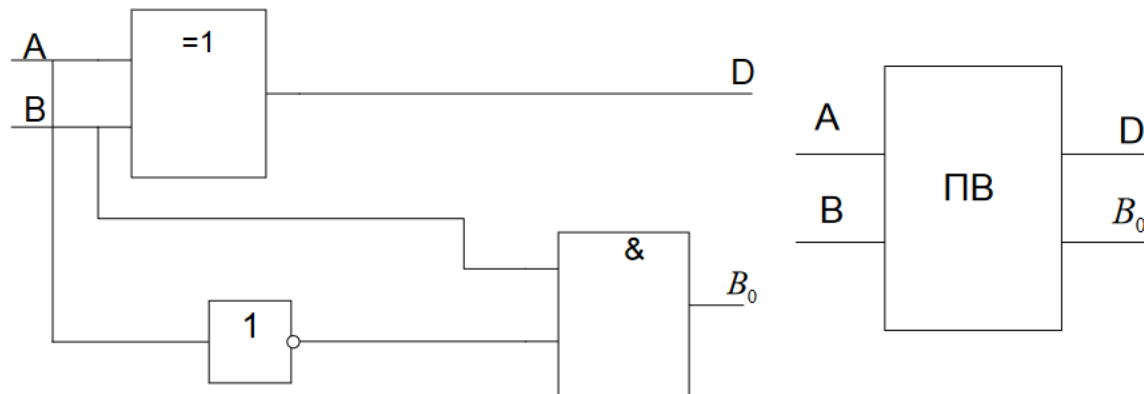
# Комбинационные схемы

## Полувывчитатель

A	B	D	$B_0$
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

$$D = A \oplus B$$

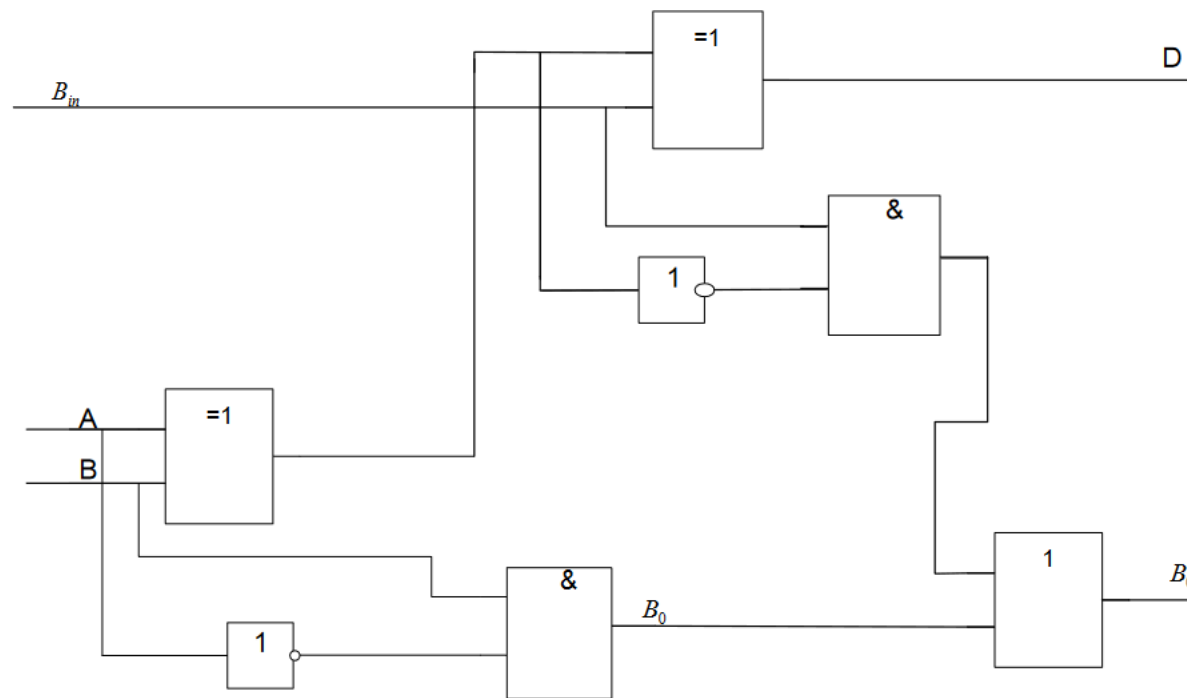
$$B_0 = \overline{A} * B$$



# Комбинационные схемы

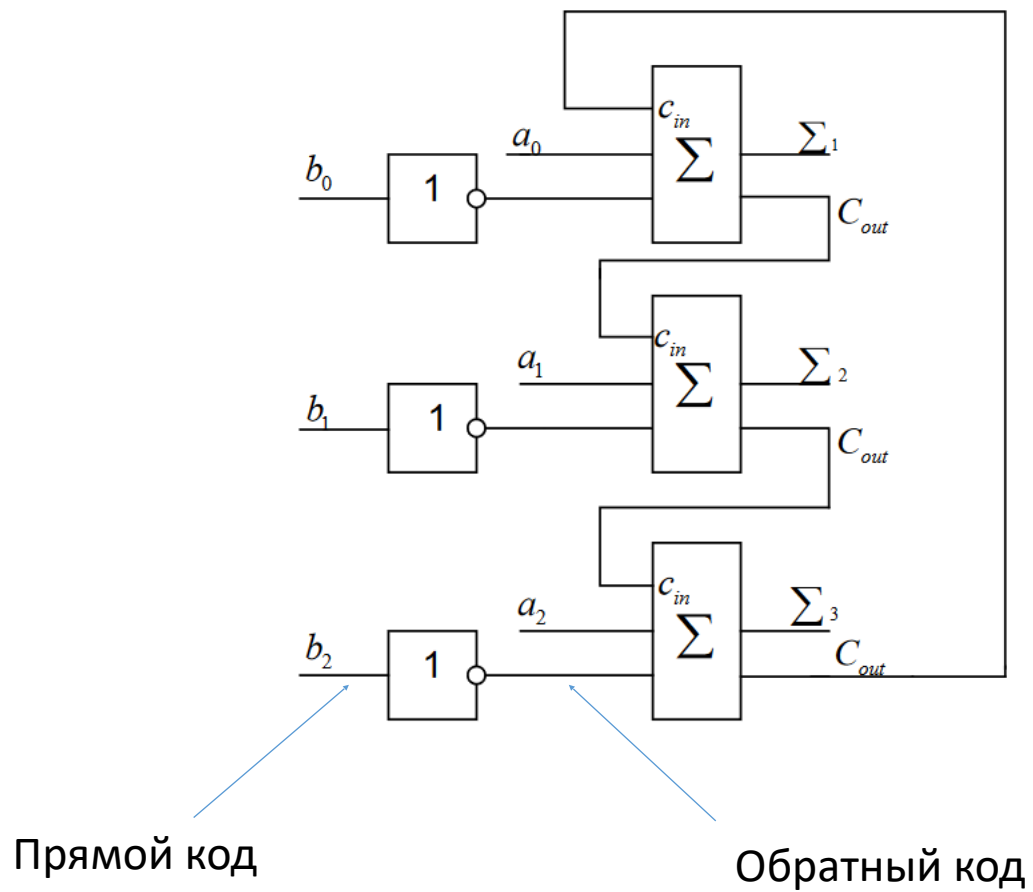
## Полный вычитатель

A	B	$B_{in}$	D	$B_0$
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1



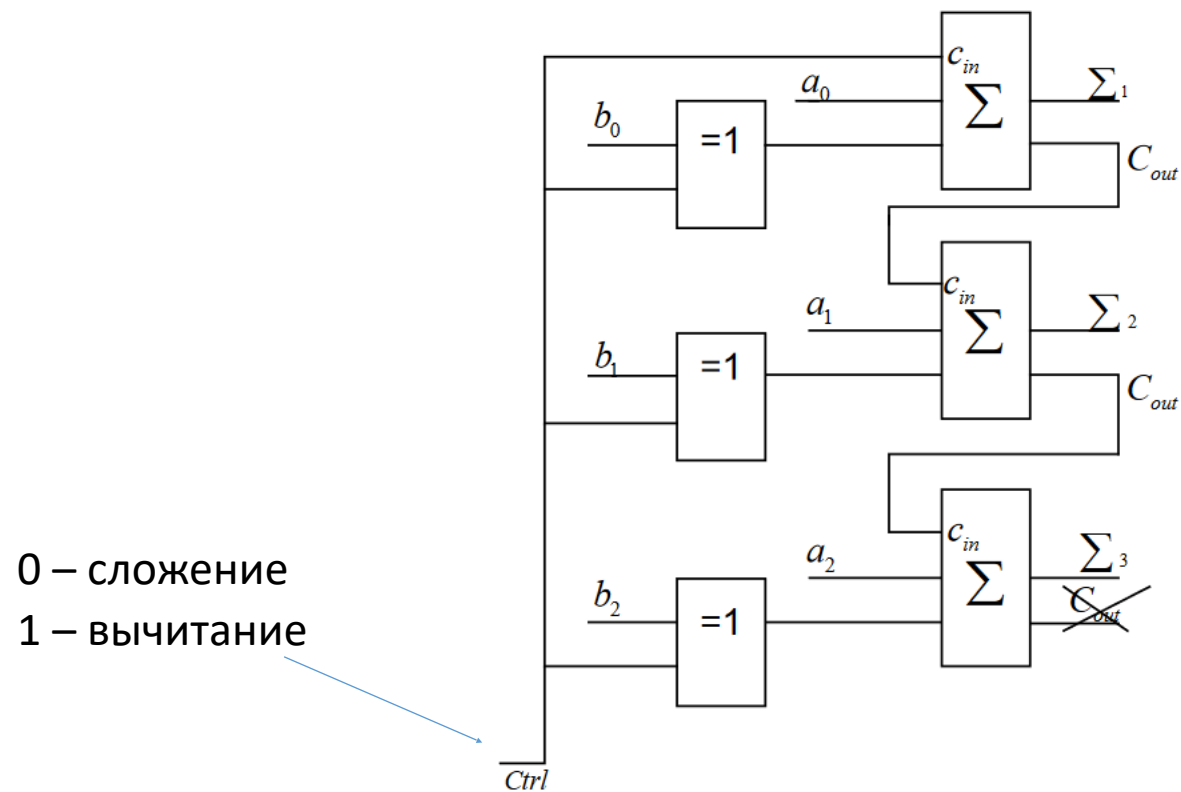
# Комбинационные схемы

## Многоразрядный вычитатель с переводом в обратный код



# Комбинационные схемы

## Универсальный сумматор-вычитатель в дополнительном коде







# Содержание лекции



- Комбинационные схемы
  - Синтез универсальных комбинационных схем с помощью карт Карно
  - (Де)кодеры и (де)мультиплексоры
- Последовательностные схемы
  - Триггеры
  - Регистры и счетчики
- Аппаратная реализация арифметических операций
  - Сумматоры и вычитатели
  - **Умножители**

# Последовательностные схемы

## Умножители

- Табличные
- Многократное сложение
- Сложения и сдвиги
- Модульные

# Последовательные схемы

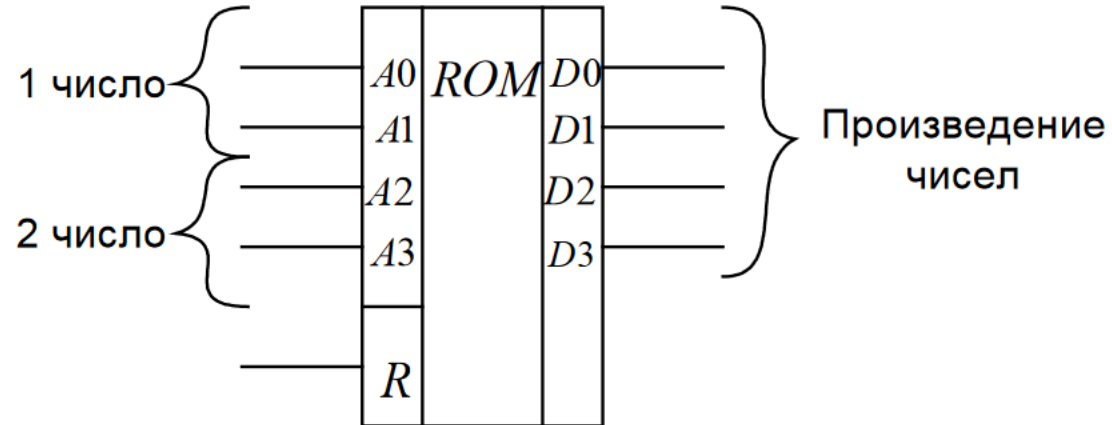
## Умножители

- **Табличные**
- Многократное сложение
- Сложения и сдвиги
- Модульные

$A_2A_3   A_0A_1$	00	01	10	11
00	0	0	0	0
01	0	1	2	3
10	0	2	4	6
11	0	3	6	9

Преимущества: высокая скорость

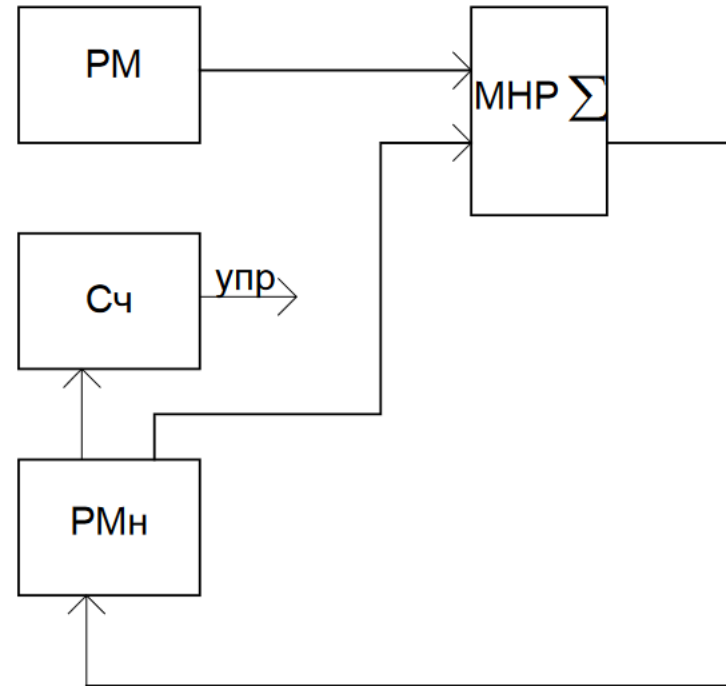
Недостаток: применим только для чисел небольшой разрядности.



# Последовательные схемы

## Умножители

- Табличные
- **Многократное сложение**
- Сложения и сдвиги
- Модульные



PM-регистр множимого  
Сч-счетчик  
PMн-регистр множителя

Преимущества: простота схемной реализации

Недостаток: медленная скорость выполнения операций

# Последовательностные схемы

## Умножители

- Табличные
- Многократное сложение
- **Сложения и сдвиги**
  - **Умножение «столбиком»**
- Модульные

# Последовательностные схемы

## Умножители

Метод основан на использовании представления числа в виде степенного

- Табличные ряда.
- Многократное сложение
- Сложения и сдвиги
- **Модульные**

$$A = \sum_{i=0}^{n-1} a_i * 2^i, \quad B = \sum_{j=0}^{n-1} b_j 2^j$$

$$A * B = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} a_i * b_j * 2^{i+j}$$

Операция суммирования и умножения на степень числа 2 осуществляется специальным параллельным сумматором, который также может быть реализован в виде комбинационной схемы.

Модульное умножение может быть осуществлено над числами в параллельном коде и над числами в последовательном коде.

Преимущество: высокая скорость.

Недостаток: существенная сложность.